PHOTOGRAPH THIS SHEET

AD A096245

DTIC ACCESSION NUMBER

II

LEVEL

I

INVENTORY

LOCKHEED MISSILES AND SPACE CO., INC., PALO ALTO, CA. PALO ALTO RESEARCH LAB.

PASSIVE IMAGERY NAVIGATION. FINAL TECHNICAL REPT. 15 SEP 78-15 DEC. 80. 15 DEC. 80. REPT.NO. LMSC-D767313 CONTRACT F33615-78-C-1612 AFWAL-TR-81-1044

DOCUMENT IDENTIFICATION

DISTRIBUTION STATEMENT A

Approved for public release;
Distribution Unlimited

DISTRIBUTION STATEMENT

ACCESSION FOR	
NTIS GRA&I	☒
DTIC TAB	□
UNANNOUNCED	□
JUSTIFICATION	
BY	
DISTRIBUTION /	
AVAILABILITY CODES	

DIST	AVAIL AND/OR SPECIAL	
A		

DISTRIBUTION STAMP

DATE ACCESSIONED

81 3 11 076

DATE RECEIVED IN DTIC

PHOTOGRAPH THIS SHEET AND RETURN TO DTIC-DDA-2

AFWAL-TR-81-1044

PASSIVE IMAGERY NAVIGATION

Palo Alto Research Laboratory
Lockheed Missiles & Space Company, Inc.
Palo Alto, California 94304

Advanced Research Projects Agency
1400 Wilson Boulevard
Arlington, Virginia 22209

December 1980

Technical Report AFWAL-TR-81-1044
Final Report for Period 15 September 1978 - 15 December 1980

Approved for public release; distribution unlimited.

Avionics Laboratory
Air Force Wright Aeronautical Laboratories
Air Force Systems Command
Wright-Patterson Air Force Base, Ohio 45433

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed by the Office of Public Affairs (ASD/PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

Louis A Tamburino

LOUIS A. TAMBURINO
Project Engineer

John O Mysing

JOHN O. MYSING, Tech Mgr
Information Presentation
& Control Group
Information Processing Technology
Branch

FOR THE COMMANDER

Raymond E Siferd

RAYMOND E. SIFERD, Colonel, USAF
Chief, System Avionics Division
Avionics Laboratory

"If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify AFWAL/AAAT *, W-PAFB, OH 45433 to help us maintain a current mailing list".*

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFWAL-TR-81-1044	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) PASSIVE IMAGERY NAVIGATION		5. TYPE OF REPORT & PERIOD COVERED Final Technical Report 9/15/78 to 12/15/80
		6. PERFORMING ORG. REPORT NUMBER LMSC-D767313
7. AUTHOR(s) O, Firschein, Principal Investigator M. J. Hannah D. L. Milgram, and C. M. Bjorklund		8. CONTRACT OR GRANT NUMBER(s) F33615-78-C-1612
9. PERFORMING ORGANIZATION NAME AND ADDRESS Palo Alto Research Laboratory Lockheed Missiles & Space Company, Inc. Palo Alto, CA 94304		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS ARPA Order No. 3608
11. CONTROLLING OFFICE NAME AND ADDRESS Advanced Research Projects Agency 1400 Wilson Boulevard Arlington, Virginia 22209		12. REPORT DATE December 15, 1980
		13. NUMBER OF PAGES 145
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Air Force Wright Aeronautical Laboratories (AFSC) Avionics Laboratory AFWAL/AAAT System Avionics Division Wright-Patterson AFB, Ohio 45433		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Key Words:
Passive Navigation, Dead Reckoning, Image-Based Navigation, Landmark Analysis, Stereo Analysis, Bootstrap Navigation, Image Understanding, Image Segmentation, Digital Image Processing

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This technical report describes the Passive Navigation study, an investigation to determine if a small, low-flying aircraft can be navigated using passively sensed images of the ground. The system described uses a position determination concept based on motion stereo; i.e., a stereo pair derived from a single camera sensing the scene at two different points on the flight path. A bootstrapping approach to position determination has been developed that starts with a set of known ground points and projects points forward from frame to frame. This stereo bootstrapping system forms the major portion of the system, and was a major portion of the study.

The second important subsystem is the Landmark Subsystem, used to correct the calculated position of the vehicle after a number of bootstrap iterations have been performed. The landmark subsystem is an edge analysis approach that associates edges into symchains, robust edge sequences that can be matched symbolically against a reference set of symchains for a known landmark.

The report discusses various aspects of the bootstrapping procedure, including the major components, the mechanics of operation, and an error analysis obtained by simulation. Results using images obtained from the U.S. Army Night Vision Laboratory terrain model are shown in the report. The report concludes with suggestions for future work in passive navigation.

FOREWORD

The concept of an image-based autonomous navigation system that mimicks human performance originated with Lt. Col. D. L. Carlstrom, USAF, while he was Program Manager at DARPA. Since 1978, the study of Passive Imagery Navigation based on this concept has been carried out by the Lockheed Palo Alto Research Laboratory as part of the DARPA Image Understanding Program under Lt. Col. L. E. Druffel, USAF. The study has been monitored by L. A. Tamburino, Air Force Avionics Laboratory, Wright-Patterson AFB. The study was carried out by a number of researchers within the Signal Processing Laboratory, J. J. Pearson, Manager. M. J. Hannah is responsible for the bootstrap stereo concept and the development of the stereo subsystem; C. M. Bjorklund and D. L. Milgram developed the landmark subsystem; and O. Firschein was the Principal Investigator and systems integrator. The Stanford University Artificial Intelligence Center subcontract was directed by Prof. T. O. Binford.

The cooperation of the U.S. Army Night Vision Laboratory in the use of their terrain model is gratefully acknowledged.

The views and conclusions contained in this document are those of the authors, and should not be interpreted as necessarily representing the official policies either expressed or implied, of the Defense Advanced Research Projects Agency of the U.S. Government.

SUMMARY

- Task Objective

The objective of the Passive Imagery Navigation study is to determine if a small, low-altitude, slow-flying vehicle can be autonomously navigated to a target using passively sensed images of the ground (Section 1).

- Technical Problems

There are three main problems to be solved:

(1) Position Extrapolation By comparing sequential image frames obtained from a moving vehicle, it is possible to determine V/H, where V is the ground velocity and H is the relative altitude above the ground. (Note that the altimeter gives the absolute altitude above sea level.) The determination of H can be done by stereo analysis. Once V is determined, the position of the vehicle can be obtained by integration (section 2.3).

(2) Checkpoint Correction. Because of errors in obtaining V, a correction to the extrapolated position must be made periodically using known landmarks on the ground. The matching of sensed landmarks to stored reference landmarks is a major problem due to variations between sensed and stored data due to time of day, illumination, weather, and season (section 2.4).

(3) Subsystem Interaction. In addition to the basic subsystems indicated above, the proper interaction between subsystems must be provided to deal with loss of images in a sequence and to deal with subsystem failure (section 2.2).

- General Methodology

The original concept for vehicle position determination was to obtain V/H by correlation and obtain H by using motion stereo. As the stereo subsystem was

developed, it was realized that it could serve as a means of vehicle location, and the V/H and H approach was abandoned. Dead reckoning is still performed as a backup system by using the wind triangle and instrument readings (section 2.5).

The present system uses a position determination concept based on motion stereo, i.e., a stereo pair derived from a single camera sensing the scene at two different points on the flight path. (Typically, an overlap of 75% is required.) A "bootstrapping" approach to position determination has been developed (section 3.2) that starts with a set of known ground points and projects points forward from frame to frame. At each image-sensing time, known points are used to compute the present camera orientation, and two consecutive camera models are then used to locate new points on the ground. The stereo system forms the major portion of the system, and was a major portion of the study. The report discusses various aspects of the bootstrapping procedure, including the major components, the mechanics of operation, and an error analysis obtained by simulation (section 3.3). A description of the videotape that was made to demonstrate the bootstrapping process is also given. This demonstration used images taken from the U.S. Army Night Vision Laboratory terrain model.

The second major subsystem is the Landmark Subsystem (Section 4) used to correct the calculated position of the vehicle after a number of bootstrap iterations have been performed. The landmark subsystem, based on work performed under the Lockheed Independent Research Program, is an edge analysis approach that associates edges into "symchains," robust edge sequences that can be matched symbolically against a reference set of symchains for a known landmark (section 4.2). Several examples of landmark matching are shown in the report (section 4.4).

- <u>Technical Results</u>

Images were gathered using a video camera that was flown over the U.S. Army Night Vision Laboratory terrain model at Ft. Belvoir, Va. The videotape was then digitized for storage in the minicomputer image memory for subsequent processing. The following technical results were obtained:

- A new operator for determining 'interesting' portions of an image was developed (section 3.1.2)
- The bootstrap stereo concept of navigation was developed and programmed (section 3.2)
- The handoff of points from frame to frame in the bootstrapping process was demonstrated
- An error simulation of the bootstrap approach showed that a match accuracy of about 1/10 pixel is sufficient to achieve a large number of iterations before divergence occurs (section 3.3)
- The error analysis showed the importance of using a wide-angle lens, with the camera pointing in the backward direction (section 3.3)
- A landmark matching approach based on identifying long chains of intensity gradient edges was developed and demonstrated (Section 4)
- The interactions between subsystems were delineated (Section 2)
- An approach to ground velocity determination based on optical flow, and using a linear solid state sensor was described (section 2.5 and Appendix B)

● Important Findings and Conclusions

The key findings were:

- The bootstrap stereo concept for position determination was demonstrated, and an error simulation showed that the bootstrap approach can attain a reasonable distance before error buildup becomes excessive (section 3.3)
- Landmark determination, based on extracting and describing edges in a sensed scene and comparing them against a stored description, was shown to work in the limited experiments carried out (section 4.4)

- Special Comments

As is usual in image-based research, it was difficult to obtain calibrated image data, including both ground truth concerning the known location of ground points in the image sequence, as well as calibration of the sensor taking the images (section 6.1). While isolated frames of properly calibrated data are becoming more common, our application requires highly overlapping images (65 to 75%); sequences of calibrated imagery that meet this requirement are still not readily available.

- Implications for Further Research

Simulations showed that the concept of navigating a small, low-altitude aircraft autonomously using passively sensed images is viable (section 3.3). However, further experiments using a calibrated video camera and ground truth for both model and real terrain images are required before one can be certain of the practicality of the approach (section 6.1). In addition, the following conceptual developments are required:

- The landmark subsystem should incorporate multiple techniques to achieve more effective landmark determination (section 6.3). As discussed in section 4.1, the landmark database can indicate which technique should be used for each stored landmark
- The stereo system computation time should be decreased by taking advantage of parallelism, and by using the calculated velocity to aid in the determination of the corresponding points in the stereo pair (section 6.2)
- The present system does not incorporate strategies for use when the navigation system gets confused or lost due to loss of data or of subsystem operation. Future research should incorporate the type of expert judgment used by a navigator for such situations, e.g., "follow a major highway or a river until a known landmark is found."

CONTENTS

ILLUSTRATIONS

TABLES

Section 1
INTRODUCTION

Automatic navigation of a subsonic vehicle (80 to 200 mph) flying at a low altitude (1500 to 5000 ft above the ground) is of great interest for unmanned flight applications such as small aircraft used for reconnaissance. For a considerable period of time during such a mission, and preferably for its entire duration, it is desirable to use a minimum of communication between the ground station and the vehicle. This requirement necessitates an autonomous mode of operation, including navigation to, and over, the target areas. A simple pre-programmed flight plan is subject to gross errors due to unpredictable winds and turbulence which cannot be sensed onboard the vehicle, since such inexpensive reconnaissance vehicles are not equipped with inertial guidance systems. However, these vehicles do contain attitude-sensing devices such as a vertical gyro, a gyrocompass, and, sometimes, also rate gyros. In addition, there is always a barometric altimeter and an airspeed sensor in the onboard instrumentation package. Microprocessors are also becoming standard items providing an ever-growing computational facility onboard the vehicle.

Recently, there has been interest in using sensed images of the terrain as the basis for an autonomous navigation system. This interest has been sparked by the fact that the pilot of a small aircraft uses visual sensing to a large degree, monitoring flight progress against a flight plan graph, Fig. 1-1. Basically the flight plan graph consists of a line representing the flight plan time from departure to destination or turning point and roughly paralleling the true course of a flight. Predicted times to various points along the true course, together with departure and destination times, are plotted on this time scale. Thus, the flight plan graph represents a visual time line comparable to the predicted track. Using the time line, the estimated time of arrival to any point on the predicted track can be determined. Comparison with a fix, checkpoint, or obstacle gives the aviator or observer an indication of whether he is ahead or behind his flight plan, and whether he is on course.

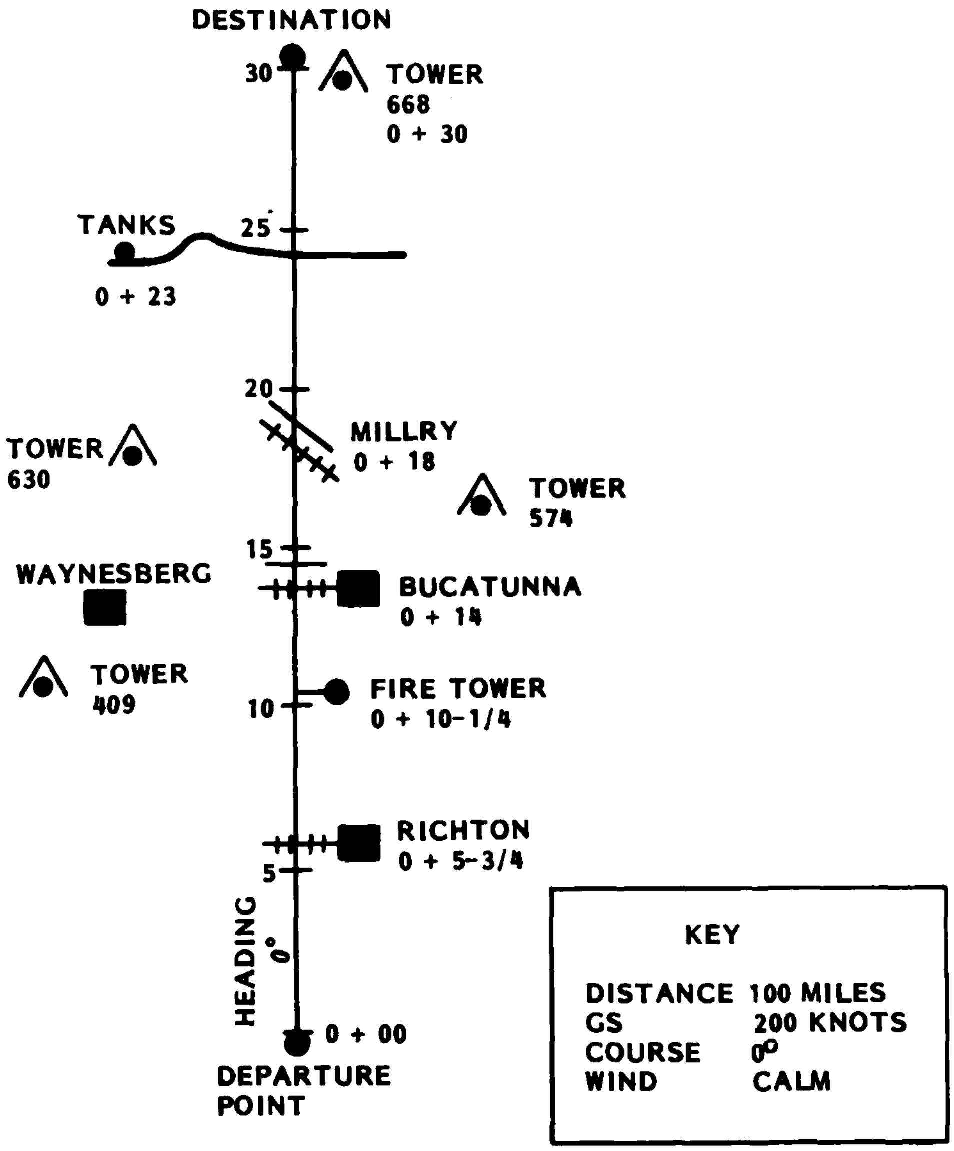


Fig. 1-1 Flight Path Graph

It is hoped that the availability of small, solid state sensors and microprocessors coupled with advances in image understanding research will enable us to simulate the human's behavior.

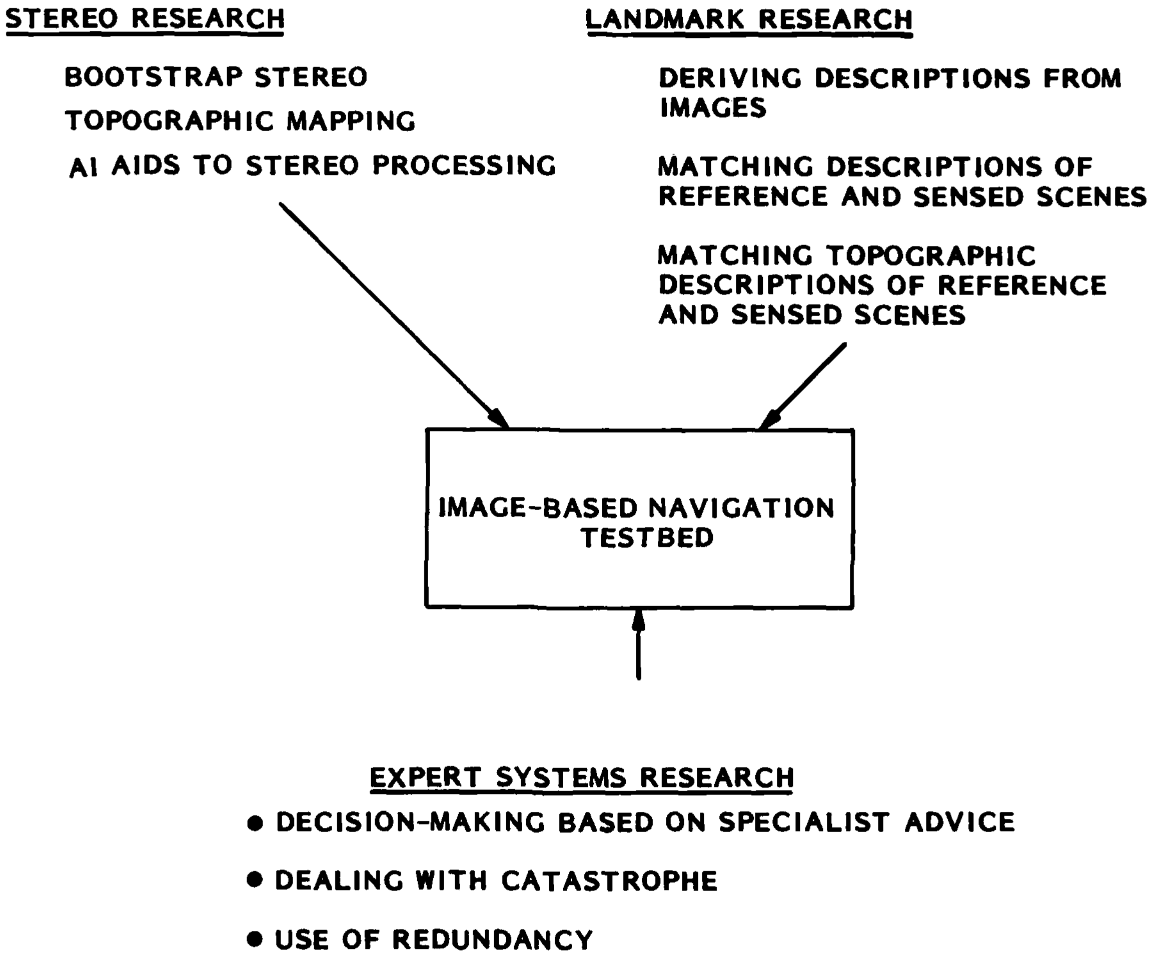
The Passive Navigation Study, carried out for the past 2 years by the Lockheed Palo Alto Research Laboratory, considers an image-based navigation system that acts as a testbed for integrating various image understanding research topics (Fig. 1-2).

The major topics are:

- Stereo Research – Investigation of the use of a time-sequence of images to determine the location of a vehicle, and to construct a topographic depth map periodically
- Landmark Research – Development of methods for deriving descriptions from sensed images and for comparing these descriptions against stored reference descriptions
- Expert Systems Research – Determination of ways of using information from the various subsystems to deal with failure of a subsystem or unavailability of images for a period of time

By allowing the subsystems to interact, and by providing the subsystem with "reasonableness" tests that indicate subsystem failure, it is possible to achieve more robust systems performance.

The study postulates an autonomous aerial vehicle that uses passively sensed images taken by an onboard camera as well as measurements from conventional instruments in order to navigate as does the human pilot. The autonomous aircraft has redundant and independent subsystems, so that partial or complete failure of a subsystem does not cause the overall system to fail. Thus, we develop an "executive" computer program that has available "specialist" subsystems, some of which are capable of analyzing images of the terrain. Each subsystem is called on by the Executive for a specific task, and reports back to the Executive



STEREO RESEARCH

BOOTSTRAP STEREO

TOPOGRAPHIC MAPPING

AI AIDS TO STEREO PROCESSING

LANDMARK RESEARCH

DERIVING DESCRIPTIONS FROM IMAGES

MATCHING DESCRIPTIONS OF REFERENCE AND SENSED SCENES

MATCHING TOPOGRAPHIC DESCRIPTIONS OF REFERENCE AND SENSED SCENES

IMAGE-BASED NAVIGATION TESTBED

EXPERT SYSTEMS RESEARCH

- DECISION-MAKING BASED ON SPECIALIST ADVICE
- DEALING WITH CATASTROPHE
- USE OF REDUNDANCY

Fig. 1-2 Image-Based Navigation Considered as a Testbed for Research Projects

with the results and with associated confidence estimates. Just as in the case of a human executive, the Executive Program must be able to decide between conflicting reports from the subsystems.

To date, the Passive Navigation Study has concentrated on the Stereo Subsystem and on defining the interactions among the Executive, the Stereo Subsystem, and the Landmark Subsystem. The following is a summary of the results of the study.

- System Aspects. The interaction of the various subsystems has been defined as described in Section 2. Of particular interest is the redundancy in vehicle location obtained from both the stereo and the dead reckoning subsystems. Both subsystems provide a confidence measure for the location estimate, with the stereo estimate based on computation residuals and the dead reckoning estimate based on the "age" of the wind velocity estimate.
- Stereo Subsystem. The stereo system described in Section 3 is the focus of the navigation system, since it not only performs the "bootstrap" navigation, but also can obtain the relative altitude of the vehicle. It has the advantage of being independent of aircraft orientation, since its camera model solver can deal with a camera that is not aimed at the nadir point. The automatic handoff of points from frame to frame in the bootstrap operation is complete, and has been described in detail in an article in the April 1980 Image Understanding Workshop Proceesings (Ref. 1). Error simulations using synthetic data have been used to determine the best combinations of look-angle and field-of-view so as to maximize the total distance that can be bootstrapped before a landmark fix is required.
- Landmark Subsystem. The role and interactions of the Landmark Subsystem have been identified using an intensity-based landmark system developed under the Lockheed Independent Research Program. This approach identifies edges of objects and finds sets of connected edge points, called symchains. The symchains are represented symbolically in terms of the line segments and the angles between them, and this description is then compared with reference symchain descriptions in the

landmark database. Experiments conducted using the NVL imagery show increased promise with the incorporation of a global matching algorithm.

These topics are discussed in more detail in the remaining sections.

Section 2
CONCEPTUAL FRAMEWORK

This section provides the conceptual framework for the Passive Navigation Study. The Image-Based Navigation system is first described, followed by discussions of the Executive, the Stereo Subsystem, and the Landmark Subsystem.

2.1 IMAGE-BASED NAVIGATION SYSTEM

To carry out the mission of autonomous reconnaissance, an Image-Based Navigation System must be developed to keep the vehicle close to the planned course and altitude. This is the purpose of the Navigation Expert, an Executive computer program that weighs evidence, makes decisions, and controls the vehicle's flight, with the help of a group of specialist subsystems that are called on for advice. As shown in Fig. 2-1, subsystems that have been identified include an Instruments Subsystem, a Dead Reckoning Subsystem, a Stereo Subsystem, and a Landmark Subsystem. Each of these provides information on its specialty along with confidence measures so that the Executive can weigh contradictory results from two or more subsystems.

The Instruments Subsystem (IS) is the simplest of the specialists. It will keep track of instrument readings and report them. The IS will also be responsible for taking the needed imagery, using a sensor fixed to the underside of the vehicle.

The Dead Reckoning Subsystem (DRS) is also rather simple. Its job is to extrapolate the current position and course from the last several known true positions, course estimates, velocity estimates, flight times, etc.

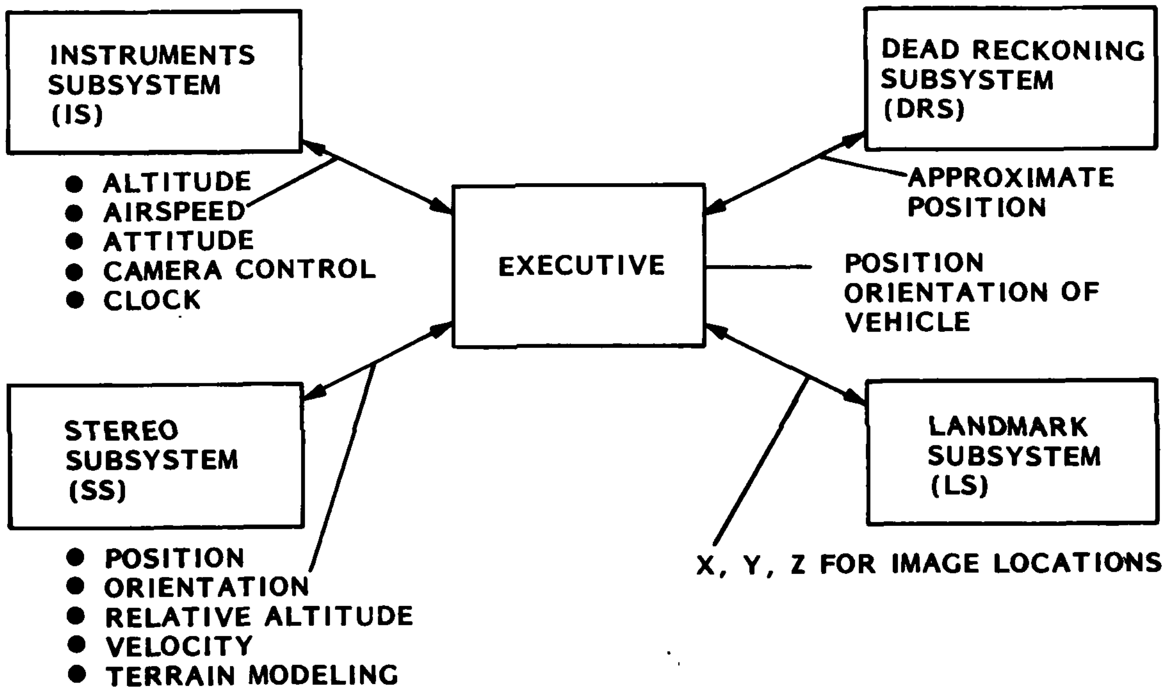


Fig. 2-1 Components of the Navigation Expert

The major subsystem, the Stereo Subsystem (StS), makes measurements on images of the underlying terrain to determine the actual position of the vehicle from the positions of the selected terrain points. It does this by using "motion stereo" in which a single camera senses the ground as the vehicle moves. The StS is initialized from a set of known landmarks from which it determines the vehicle's position for the first two images. Given these two positions, it uses stereo techniques to locate new terrain points, and from these new points it determines its next position. This bootstrapping procedure is analogous to the one used in moving across a marsh with the aid of three boards – one stands on two of the boards, while moving the third board ahead for the next step.

The Landmark Subsystem (LS) has the duty of recognizing landmarks along the desired course. The LS is required to initialize the StS. Also, since both the StS and the DRS are subject to accumulated errors, landmark checkpointing is needed to correct the position.

The detailed interaction is shown in Fig. 2-2 and described below.

2.2 ROLE OF THE EXECUTIVE

The role of the Executive is to coordinate the results provided by the specialist subsystems while arbitrating any differences of opinion they may have. As an example, let us assume that the Executive has decided to rely primarily on the StS for its position information. The interaction among the modules would proceed as follows.

The Executive first obtains an image from the Instruments Subsystem and passes control to the Landmark Subsystem, asking the LS to locate landmarks in the image which can be matched to their stored reference representations.

Assuming that the LS is successful, it returns a list of image points and corresponding 3-D terrain locatiçns. The Executive would then give the

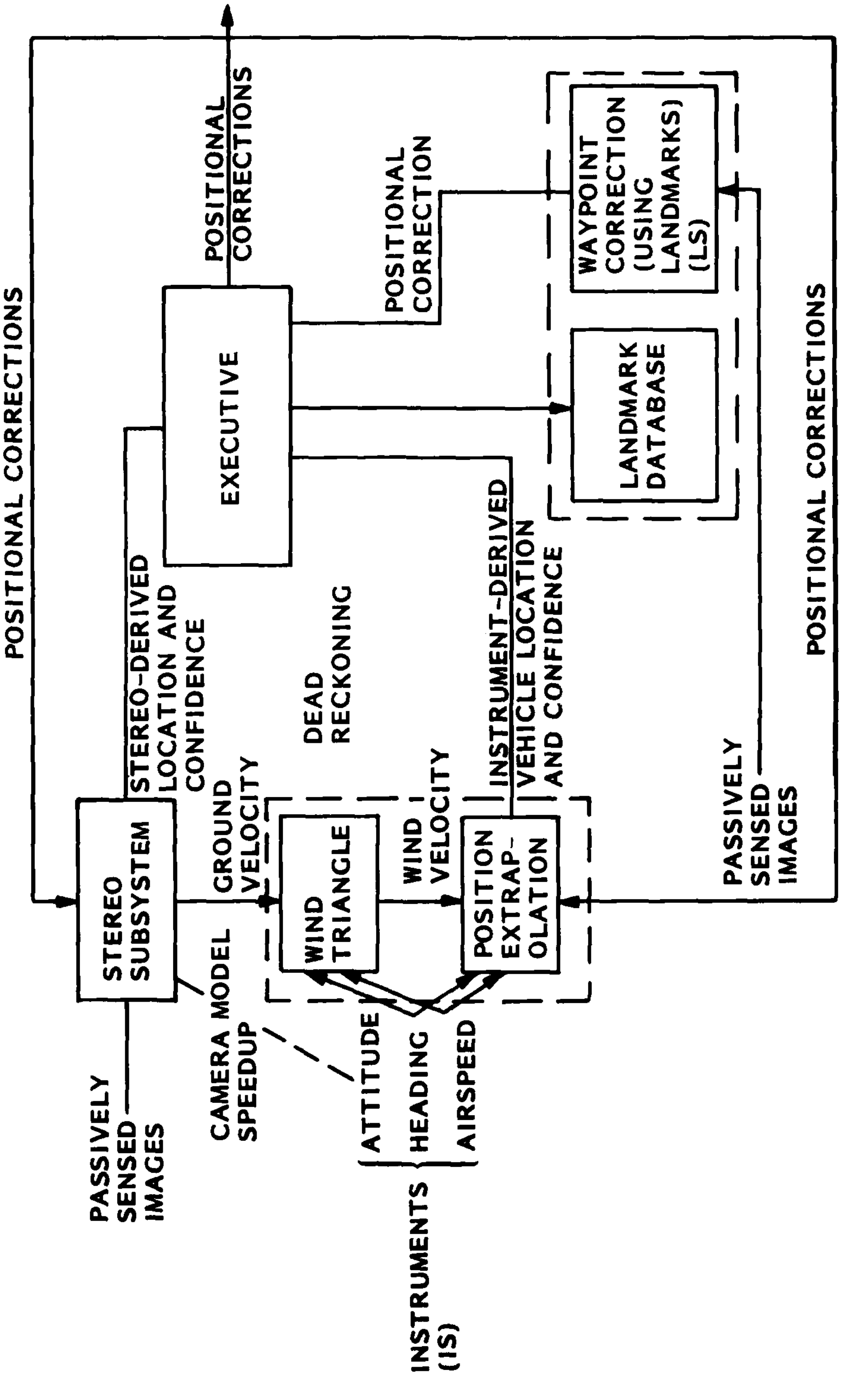


Fig. 2-2 Interaction of the Various Subsystems

landmarks list to the StS in the initialization mode and would receive a position/orientation opinion for that image. The Executive is now ready to use the StS in the bootstrapping mode.

At each iteration, the Executive obtains a current image from the Instruments Subsystem. This, along with the last image, provides the two images that the StS needs. The Executive now gives the StS the current landmarks list, which originated with the Landmark Subsystem and may have been updated by previous bootstrappings of the StS. The Executive also gives the StS the position/orientation data for the last image. From the Instruments Subsystem, the Executive gets the camera orientation data corresponding to the current image; this, with a current position estimate from the Dead-Reckoning Subsystem, gives the StS its current position/orientation estimate.

After doing its calculations, the StS returns to the Executive with its position/orientation opinion, its confidence measures, and its updated landmark list. The StS can also provide terrain elevations for the Landmark Subsystem. The Executive supplies the Dead Reckoning Subsystem with the new position, and requests an extrapolation of the current position, speed, and course toward the checkpoint. Based on this, the Executive applies any course corrections that may be needed and initiates another round of bootstrapping.

As a checkpoint is approached, the Landmark Subsystem is again invoked to search for the recognition landmarks. When the LS is successful, these new landmarks are used to reinitialize the StS.

It is anticipated that the specialist subsystems will be implemented as completely independent processes or pieces of hardware. Thus, it will be possible to have the StS continue to bootstrap while the LS looks for new landmarks. Of course, at the same time, other independent specialists, such as the Dead Reckoning Subsystem, can be constantly processing their data and giving their opinions. The Executive receives all of these opinions and uses the combination of them that seems most "reasonable."

2.3 STEREO SUBSYSTEM

If two pictures of the ground are taken, each at a different but known vehicle location, and the pictures overlap with ground points in common, then it is possible to perform automatic stereo analysis to determine the location of the common points. Alternatively, we can determine the aircraft position in space if we know the position and the elevation of points on the ground. These types of stereo analyses of images have been done automatically by computer for mapping purposes, but have not been previously used as the basis of a navigation system.

A major factor in this analysis is the determination of "corresponding points," i.e., where points on the ground appear in each image. Typically, one uses a correlation matching procedure to find the corresponding points for the "interesting" points (those that have brightness characteristics which stand out in some way).

A location-finding approach called "bootstrap stereo" (Ref. 1) plays a major role in the system. The idea is based on starting with known ground points to locate the airplane in space, and then using two aircraft locations to locate previously unknown points on the ground. The basic concept is discussed in detail in section 3.2.

In bootstrapping, the StS assumes that the vehicle is flying along taking occasional approximately downward-looking imagery of the terrain below it. To function properly, the StS requires that the imagery be properly focused and unobscured by clouds, and that successive frames overlap about 75%.

In the bootstrap mode, the StS requires several inputs, including two images (one each from the current and previous positions), a list of established points (their 3-D terrain positions and their film-plane locations in the previous image), the 3-D position and orientation of the camera (hence also the orientation of the vehicle to which it is fixed) at the time of the previous image, and a confidence measure for these quantities. An updated list of established points with their 3-D terrain positions and their film-plane locations in the current image is prepared, and the mean altitude of the vehicle above the terrain is sent to the Executive.

If one or more images is unavailable due to cloud cover or temporary camera failure, then the following alternatives exist for the bootstrap system: (1) it can wait for the Landmark Subsystem to provide a new group of starting points, or (2) it can use dead-reckoning estimates to make assumptions about the coordinates of a set of points, and bootstrap from these assumed points.

In either case, the Executive is informed of the problem so that the resulting location estimates can be evaluated.

2.4 LANDMARK SUBSYSTEM

The goal of a landmark subsystem is to provide position verification and correction to the navigation system based on pre-stored landmark descriptions. Intuitively, a landmark is a unique set of lines or regions that has a high probability of being detected in an image regardless of sun angle, weather, and seasonal conditions which do not obscure the view. Landmarks obviously depend on vehicle altitude and the sensor used, but at 1500 to 3000 ft, highways, crossroads, rivers, towns, and lakes are likely candidates. Landmarks should be semantically meaningful and not accidental features.

The nature of desirable landmarks for a human pilot is indicated in Table 2-1. Unfortunately, automatic computer processing of an image is far from being able to deal with such landmarks. Instead, one must rely on landmarks such as:

- Roads and rivers that show up as ribbons of parallel lines
- Homogeneous regions such as lakes
- Distinctive contours such as shorelines
- Large manmade objects such as airfields

Table 2-1

GOOD AND POOR VISUAL CHECKPOINTS FOR THE HUMAN NAVIGATOR

GOOD CHECKPOINTS	POOR CHECKPOINTS
MOUNTAINOUS AREAS	
Prominent peaks, cuts and passes, gorges. General profile of ranges, transmission lines, railroads, large bridges over gorges, highways, lookout stations. Tunnel openings and mines. Clearings and grass valleys.	Smaller peaks and ridges, similar in size and shape.
COASTAL AREAS	
Coastline with unusual features. Lighthouses, marker buoys, towns with cities, structures.	General rolling coastline with no distinguishing points.
SEASONAL CHANGES	
Unusually shaped wooded areas in winter. Dry river beds if they contrast with surrounding terrain. Dry lakes.	Open country and frozen lakes in winter unless in forested areas. Small lakes and rivers in arid sections of country – in summer – when they may dry up. Lakes (small) in wet seasons in lake areas, where ponds may form by surface waters.
HEAVILY POPULATED AREAS	
Large cities with definite shape. Small cities with some outstanding checkpoint; river, lake, structure, easy to identify from others. Prominent structures, speedways, railroad yards, underpasses, rivers and lakes. Race tracks and stadia, grain elevators, etc.	Small cities and towns, close together with no definite shape on chart. Small cities or towns with no outstanding checkpoints to identify them from others. Regular highways and roads, single railroads, transmission lines.
OTHER AREAS; FARM COUNTRY	
Any city, town, or village with identifying structures or prominent terrain features adjacent. Prominent paved highways, large railroads, prominent structures, race tracks, fairgrounds, factories, bridges, and underpasses. Lakes, rivers, general contour of terrain; coastlines, mountains, and ridges where they are distinctive.	Farms, small villages rather close together, and with no distinguishing characteristics. Single railroads, transmission lines and roads through farming country. Small lakes and streams in sections of country where such are prevalent, ordinary hills in rolling terrain.
FORESTED AREAS	
Transmission lines and railroad right-of-ways. Roads and highways, cities, towns and villages, forest lookout towers, farms. Rivers, lakes, marked terrain features, ridges, mountains, clearings, open valleys.	Trails and small roads without cleared right-of-ways. Extended forest areas with few breaks or outstanding characteristics of terrain.

As described below, a landmark database specifies for each landmark not only the landmark descriptions, but also the associated processing algorithms, and their parameters. Primitive features are first extracted from the images, and then landmark identification proceeds by first performing local matching, then determining a consistent overall global matching. If an acceptable global match is obtained, the match locations in the field-of-view are used to correct the navigation estimate.

2.4.1 Landmark Database

The system Executive uses its estimate of position and view angles to determine which entries in the landmark database are viewable. A typical landmark database entry is shown in Table 2-2. Note that in addition to the detailed descriptions of the landmark, the entry also provides global descriptions, suggestions as to techniques to use for finding the landmark, and, when readily identifiable, specific x, y, z coordinates of the landmark points.

Given the information as to which database entities contain descriptions of viewable landmarks, the landmark matcher performs its analysis of the sensed image, and responds in one of the following ways.

(1) An indication that a match was not found

(2) If a match was found, the region of the image in which it was found and the general orientation of the landmark is supplied to the Executive

(3) For readily identifiable points, the correspondence between x, y, z, coordinates for these (as found in the database) and their image coordinates

The meaning of these responses to the system Executive is shown in Table 2-3. Response 2 can arise when using landmarks such as a road, if a crossroad or other distinct feature does not exist. A match of this type can only provide information as to where in the image it was found, and the road orientation. Note that, as in

Table 2-2

TYPICAL LANDMARK DATABASE ENTRY

- Real-World Coordinates
  (Used by the Executive to determine if the landmark is viewable. Used also for locating prominent points for more accurate positional updates.)
  - Bounding rectangle
  - Prominent junctions
  - Centers of landmarks
- Type of Landmark
  (Used to determine which Landmark Analysis Subsystem should be used.)
  - Shape (natural or manmade)
  - Roads/rivers
  - Topographic
- Description
  (Aids landmark matcher in determining correctness of pieces of landmark found.)
  - General: Relation of parts, or to other landmarks; length/width ratio; closed-boundary or open shape; smoothness factor
  - Detailed: Symbolic representation or image chips for correlation matching
- Processing Techniques
  (Indication of threshold settings, and which low-level processing techniques are required as determined by the process that derived this landmark entry.)

Table 2-3

LANDMARK-MATCHER RESPONSES AND THEIR MEANING

Response Type	Response	Meaning
1	Landmark not found in sensed image.	If over a long enough time period and for enough landmarks, indicates that something is radically wrong
2	Match found where expected	Indicates navigation is ok
2	Match found, not where expected	Steer to correct and try again
3	Real-world coordinates of cross-roads, intersections, etc.	Use in coordinate transformation program to find vehicle position and orientation

the case of the human pilot of a small low-flying aircraft, the appearance of a landmark where expected, even in a rough sense, indicates that the Navigation System is not in trouble.

2.4.2 Identifying Intensity Landmarks

The landmark type included in this study is the class of features discernible in intensity-based imagery (e.g., visible or infrared). Such features as river bends, road intersections, or field boundaries appear as strongly contrasting gray-level regions whose shapes facilitate position fixing. Previous work on this contract by Bjorklund and Milgram (Ref. 2) has shown that edge structures derived from symchains can be matched against reference map structures.

This symchain approach consists of the following steps: (1) image elements that are the edges of regions are identified; (2) the boundaries are tracked, resulting in a list of edge pixels and their neighbors on the boundary; (3) the best left and right neighbors for each edge pixel are then selected as link elements; (4) the resulting connected set of links are formed into symchains; (5) the symchains are represented symbolically in terms of link element length and angle; and (6) this description is then compared with reference symchain descriptions in the landmark data base.

The largest consistent aggregation of individual matches among sensed and reference primitives determines the overall mapping of the sensed imagery to the reference. Individual matches between reference landmarks of each type and their sensed primitive representations are first identified. The global phase then computes the overall mapping which accounts for the spatial distribution of the largest number of matched primitives in the sensed domain.

If minimum global match criteria are met, the Landmark Matching Subsystem informs the Executive and passes to it the identified mapping for position updating.

A detailed description of the Stereo Subsystem is given in Section 3, and of the Landmark Subsystem in Section 4.

2.5 DEAD RECKONING SUBSYSTEM

The Dead Reckoning Subsystem provides an instrument-derived vehicle location as backup to the stereo-derived location. This is obtained by combining the ground velocity (calculated by the stereo subsystem) and the heading and airspeed (read from instruments) to obtain the wind velocity. (The wind triangle is a vector triangle whose sides are the wind velocity and the vehicle's air velocity, and whose result is the ground velocity.)

Wind velocity is stored so that when ground velocity is not available from the stereo system due to missed image frames, the airspeed and heading obtained from instruments can be combined with the stored wind velocity to obtain an estimate of the ground velocity. The ground velocity is then integrated to obtain vehicle location.

An interesting method for deriving ground velocity using a linear solid state sensor was investigated and is discussed in some detail in Appendix B. The approach is related to optical flow (Ref. 3) and to motion-compensated image compression (Ref. 4). Briefly, the concept is as follows.

In the navigation application, the camera moves over a static scene. The intensity change at a particular picture element (pixel) in going from frame to frame is a function of the distance moved and of the intensity variation in the scene. If we imagine the vehicle moving a very short distance between frames, and with motion parallel to a linear sensor, then we have the situation shown in Fig. 2-3.

The intensity at sensor pixels i and (i - 1) for two time periods is shown. The intensity at pixel for the previous time period is $I_{t-\tau}(i)$, and for the present time period is $I_t(i)$. The intensity at pixel (i - 1) at the present time is $I_t(i - 1)$. The slope of the intensity line is approximately

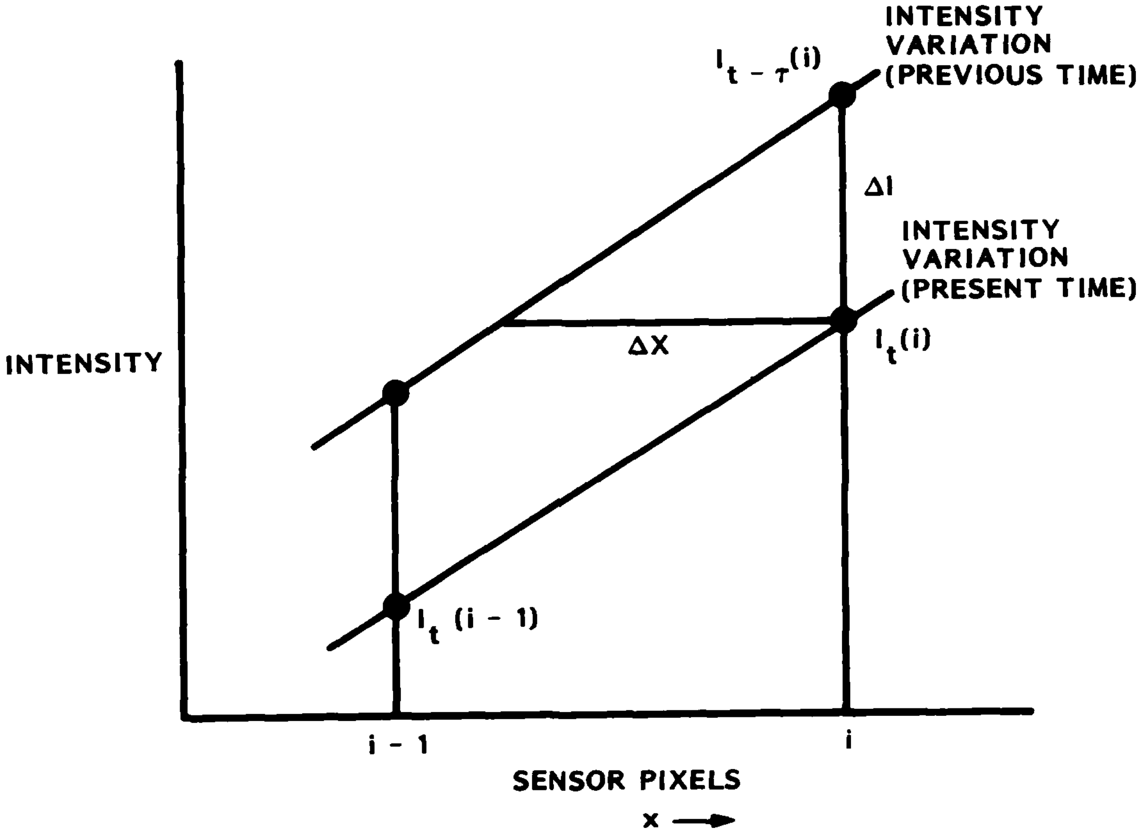


Fig. 2-3 Change in Intensity at a Sensor Pixel Due to Sensor Motion

$$\frac{dI}{dx} \approx \frac{\Delta I}{\Delta x}$$

Thus,

$$\Delta X \approx \frac{\Delta I}{\frac{dI}{dx}}$$

Substituting the intensity values, we obtain

$$\Delta X = \frac{I_t(i) - I_{t-\tau}(i)}{I_t(i) - I_t(i-1)}$$

The approximate displacement is therefore given by the ratio of the intensity difference at a pixel for two subsequent frames to the intensity difference within a frame. We can speak of this as the "time difference" divided by the "space difference" or the ratio of the "interframe" to "intraframe" difference.

Although this relationship is true only for motion in the direction of the linear sensor, Appendix B shows how one can use this concept to obtain the ground velocity of a slow-moving vehicle.

Section 3
STEREO SUBSYSTEM

The stereo subsystem of the Navigation Expert performs a variety of stereo photogrammetric tasks. The basic stereo techniques used by this subsystem are described in section 3.1. Section 3.2 explains how these techniques are combined to perform the task of stereo bootstrapping. Section 3.3 describes the error experiments and analysis performed on the bootstrap stereo components, while section 3.4 addresses the mechanization and speedup of these techniques.

3.1 STEREO TECHNIQUES

The basic requirement for stereo computation is a "stereo pair" of images, i.e., two distinct views of an object. In the Passive Navigation System, a sequence of overlapping images is obtained by a single sensor flown over the scene; these images are processed in pairs as motion stereo.

There are four basic techniques used in the stereo system:

(1) Camera Calibration – determining the camera position and orientation from known ground control points

(2) Interesting Point Selection – choosing potential new control points

(3) Point Matching – pairing a point in one image with its corresponding point in a second, overlapping image

(4) Control Point Positioning – locating points on the ground, given their positions in two images and the relevant camera positions and orientations

These techniques will be discussed in the following sections.

3.1.1 Camera Calibration

Given a set of ground control points with known real-world positions (Xi, Yi, Zi), and given the perceived locations of these points on the image plane (Ui, Vi), it is possible to determine the position (X0, Y0, Z0) and orientation (HEADING, PITCH, ROLL) of the camera which took the imagery (Refs. 5, 6). This is accomplished by a least-squares solution of a set of collinearity conditions equations, effectively minimizing the mean of the errors between each image plane point (Ui, Vi) and the projection (Ui', Vi') of that point's real-world location (Xi, Yi, Zi) onto the image. (see Fig. 3-1.) Because the equations are highly nonlinear, a solution is usually sought by iterating on linearization of the problem (Ref. 7).

This technique is somewhat sensitive to invalid points which may appear in its data set. Consequently, each camera solution is checked to see if any of the points are contributing excessively to the residual error. Such points are removed from the data set and the solution is redone, to avoid major errors. Indeed, this editing process usually has to be iterated to obtain maximum data reliability.

A promising technique under development (Ref. 8) forms analytically exact camera position and orientation models from subsets of the point data, then evaluates the points and potential models together, before refining the least-squares model from the reliable points, as above. This technique appears to be an improvement, both computationally and in terms of accuracy, to the traditional method.

3.1.2 Interesting Point Selection

The basis of stereo processing is the generalized matching of corresponding points between the two images of a stereo pair. Experience has shown, however, that some image points will match better than others, based largely on the intensity information surrounding the point. For this reason, interesting points – points with a high likelihood of being matched (Ref. 7) – are selected as the points to be matched.

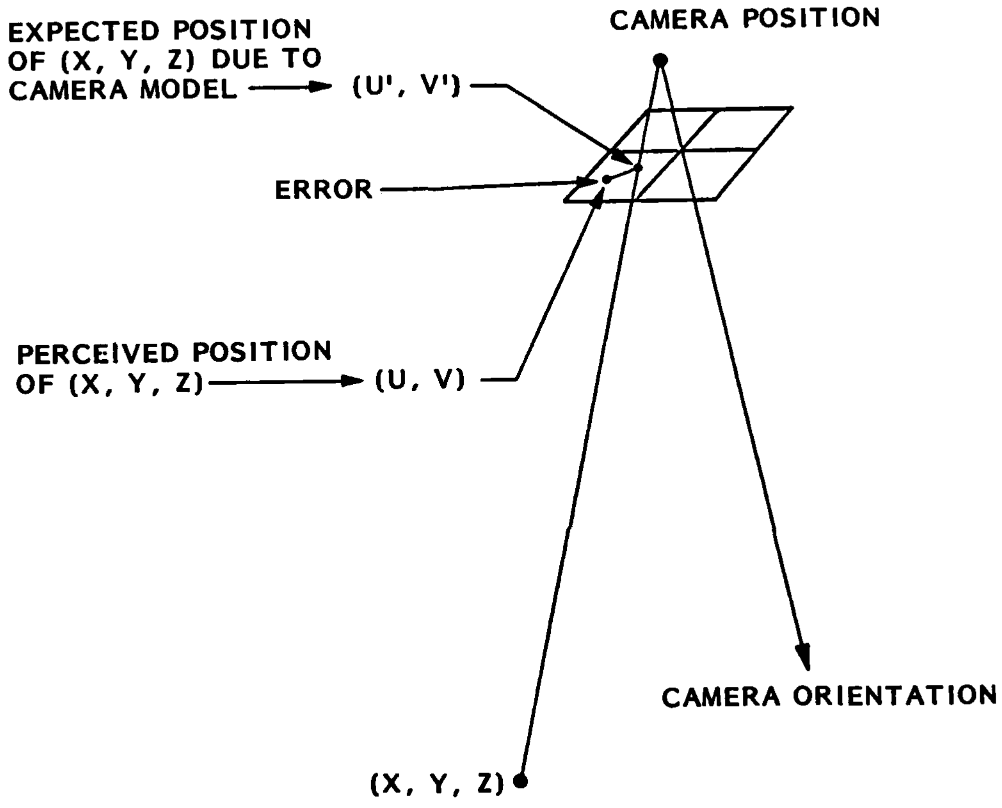

CAMERA POSITION
EXPECTED POSITION OF (X, Y, Z) DUE TO CAMERA MODEL
(U', V')
ERROR
PERCEIVED POSITION OF (X, Y, Z)
(U, V)
CAMERA ORIENTATION
(X, Y, Z)

Fig. 3-1 Calibration of Camera Position and Orientation. The position and orientation of the camera is derived by search for the camera model parameters which minimize (over a set of points) the error between the perceived position (U, V) of a point (X, Y, Z) and the position to which it would be projected (U', V') if this were the correct model

Matching is done on the basis of the normalized cross correlation between small windows of data (typically 11 × 11) around the two points in question. If the window to be matched contains little information, it can correlate reasonably well with any other area of similar low information. To avoid mismatches from attempting to use such areas, the simple statistical variance of the image intensities over the window

$$\text{var} = \underset{ij}{\text{MEAN}} \left(\text{INT}\,(i,j) - \text{MEAN}\,(\text{INT})\right)^2$$

was used as an early measure of information (Ref. 10) with only areas of high information being acceptable candidates for matching.

Matching also has trouble with strong linear edges, since an otherwise featureless area containing a strong edge will match equally well anywhere along the edge. To reject such areas, the notion of directed variance was introduced (Ref. 9). Four quantities are calculated over the window:

$$\text{dirvar1} = \text{MEAN}\,(\text{INT}\,(i,\ j) - \text{INT}\,(i + 1,\ j))^2$$

$$\text{dirvar2} = \text{MEAN}\,(\text{INT}\,(i,\ j) - \text{INT}\,(i,\ j + 1))^2$$

$$\text{dirvar3} = \text{MEAN}\,(\text{INT}\,(i,\ j) - \text{INT}\,(i + 1,\ j + 1))^2$$

$$\text{dirvar4} = \text{MEAN}\,(\text{INT}\,(i + 1,\ j) - \text{INT}\,(I,\ j + 1))^2$$

The directed variance is then defined to be the minimum of these four quantities. Points with poor visual texture will have low directed variance because adjacent samples differ little in any of the directions. Points with linear edges will show low directed variance in the direction of the edge. Conversely, points with high directed variance should avoid these defects. Thus, "interesting points" were defined to be local maxima in this "interest operator," directed variance.

We have developed another indicator of the presence of an edge in the window — the ratio of the directional variances, taken in perpendicular pairs. This measure takes advantage of the fact that a window with a strong edge will have much greater information content across the edge than along it. Because this measure does not give an indication of low information, we have combined it with ordinary variance to form an interest measure we call edged variance.

$$\text{evar} = \text{var*MIN}\left(\frac{\text{dirvar2}}{\text{dirvar1}}, \frac{\text{dirvar1}}{\text{dirvar2}}, \frac{\text{dirvar4}}{\text{dirvar3}}, \frac{\text{dirvar3}}{\text{dirvar4}}\right)$$

These measures are compared and evaluated in section 3.3.3.

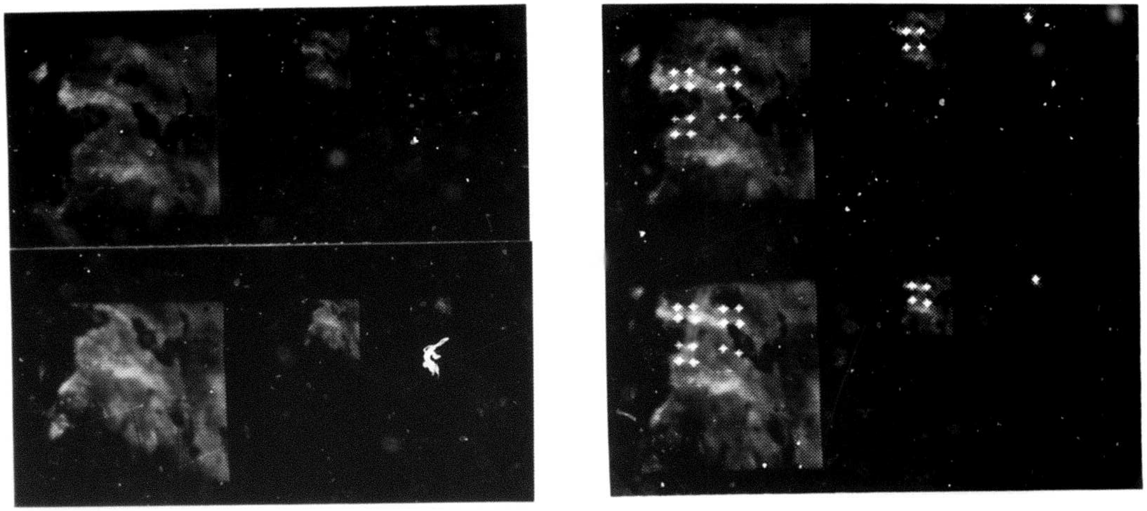
3.1.3 Point Matching

The actual matching of points in an image pair is done by maximizing normalized cross correlation over small windows surrounding the points. Given an approximation to the displacement which described the match, a simple spiraling grid search is a fairly efficient way to refine the precise match (Ref. 10). To provide that initial approximation, we have employed a form of reduction matching (Refs. 9, 10).

As shown in Fig. 3-2a, we first create a hierarchy of N-ary reduction images. Each N × N square of pixels in an image is averaged to form a single pixel at the next level. (For the example shown, N = 3.) This reduction process is repeated at each level, stopping when the image becomes approximately the size of the correlation windows being used.

Matching then begins at the smallest images. A window centered on the image is matched via the spiral search, beginning at the center of the second image. Thereafter, each matched point spawns four points around itself, offset by half a window radius along the diagonals of the window.

These are mapped down to the next level of images, carrying their parent's displacement (suitably magnified) as their suggested match approximation.



a. A hierarchy of images, each the 3 × 3 reduction of its parent

b. Matching points found by expanding a grid through the reduction image hierarchy

Fig. 3-2 Reduction Matching

These points then have their displacements refined by a spiraling search before spawning new points. This process (illustrated in Fig. 3-2b) continues until the largest images are reached, effectively setting up a grid of control points for matching.

Having this initialization, we intended that further matches be approximated from the displacement of the nearest grid control point, then refined via the spiral search. We discovered, however, that an occasional point could be lost in the process of carrying the matches down the image hierarchy, either because its match disappeared over the edge of the image, because of low information in an area, or because relief-induced distortion causes a match to be unreliable (as determined by autocorrelation thresholding (Ref. 10). Consequently, using the closest point required searching through the grid control points to determine the closest valid one.

To avoid this, we chose to approximate the displacement in a different way. Aerial imagery usually does not present any reversals in displacement, so it is reasonable to approximate the DX and DY components of the displacements by fitting polynomials in X and Y to them (Ref. 11). For each further point to be matched, its position in the first image is used to evaluate the two polynomials, producing an estimate of the position of the matching second image point.

When we used first-order polynomials for this approximation, the residual errors were on the order of a pixel, and reliable matches which had been initialized from these polynomials differed by as much as 3 pixels from the predicted displacement. Using second-order polynomials resulted in residual errors on the order of half a pixel, and reliable matches differed by less than 2 pixels from the predicted displacement. Since this was deemed adequate for initializing the local match search, higher-order polynomials were not tried.

In summary, reduction matching is used to determine approximate registration of the images and to initialize the second-order match prediction polynomials.

Matching of image points uses these polynomials to predict an approximate match, which is then refined by a local search. Autocorrelation thresholding is used to test the reliability of the match, then points are located more closely than the image grid permits by parabolic interpolation of the X- and Y-slices of the correlation values.

3.1.4 Control Point Positioning

Given the positions and orientations of two cameras and the locations of corresponding point-pairs in the two image planes, the real-world locations of the viewed ground points can easily be determined (Refs. 5, 6). The vectors from the focal points of the cameras through the respective image plane points are simply projected into space (Fig. 3-3). Since these rays rarely intersect exactly, we find their points of closest approach and average them.

If the difference is large or the real-world point is unreasonably different from its neighbors, the point is rejected as having resulted from a bad match. Otherwise, this point joins the list of control points for future processing. The resulting positions can be expressed either in real-world coordinates (absolute position) or in camera-relative coordinates (relative position).

3.2 THE BOOTSTRAP STEREO CONCEPT

Given a set of ground control points with known real-world positions, and given the locations of the projections of these points onto the image plane, it is possible to determine the position and orientation of the camera which collected the image, a process known to traditional photogrammetrists as "space resection" (Ref. 6). Conversely, given the positions and orientations of two cameras and the locations of corresponding point-pairs in the two image planes, the real-world locations of the viewed ground points can be determined, a process known as "space intersections" (Ref. 6). Combining these two techniques iteratively produces the basis for bootstrap stereo.

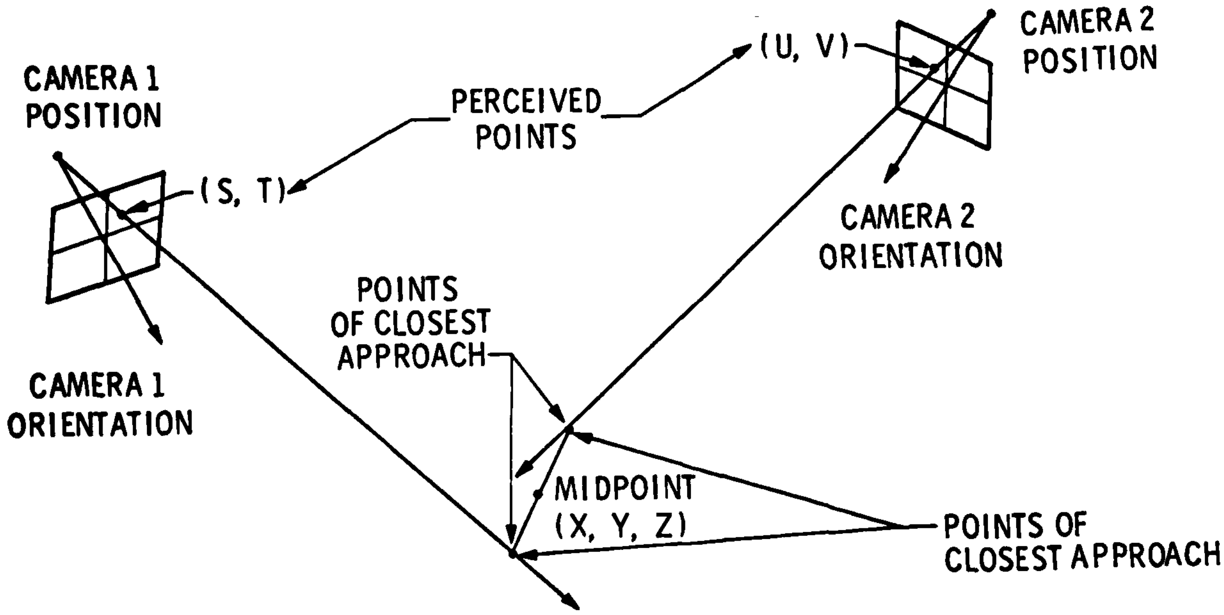


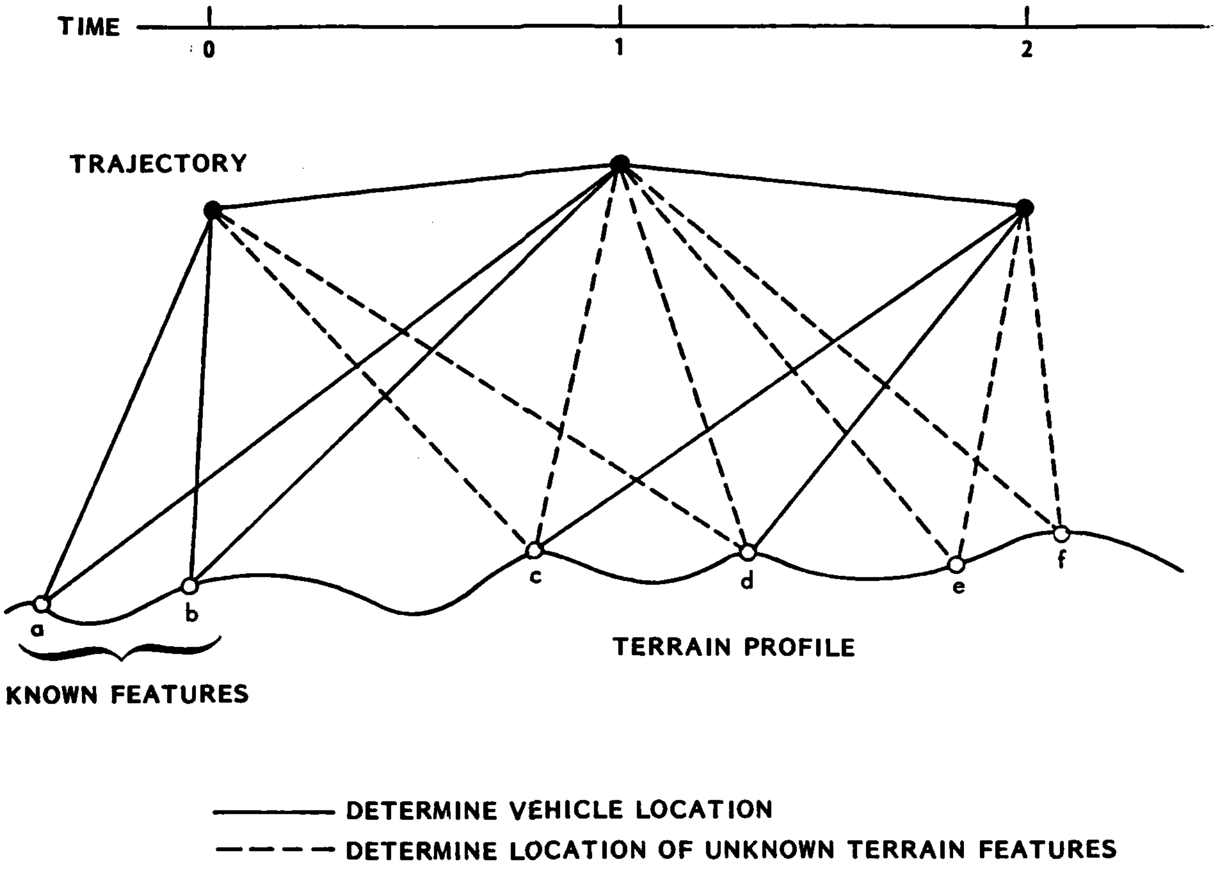
Fig. 3-3 Point Position Calculation. The points (S, T) and (U, V) are projected through their respective cameras. Their intersection (X, Y, Z) is defined to be the midpoint between the points of closest approach for these rays

Figure 3-4 shows an aircraft which has obtained images at three points in its trajectory. The bootstrap stereo process begins with the set of landmark points, a and b, whose real-world coordinates are known. (In reality, at least four points would be needed; only two are shown here to simplify the diagram.) From these, the camera position and orientation is determined for the image frame taken at Time 0. Standard image-matching correlation techniques (Ref. 10) are then used to locate these same points in the second, overlapping frame taken at Time 1. This permits the second camera position and orientation to be determined.

Because the aircraft will soon be out of sight of the known landmarks, new landmark points must be established whenever possible. For this purpose, interesting points – points with a high likelihood of being matched (Ref. 9) – are selected in the first image and matched in the second image. Successfully matched points have their real-world locations calculated from the camera position and orientation data, then join the landmarks list. In Fig. 3-4, landmarks c and d are located in this manner at Time 1; these new points are later used to position the aircraft at Time 2. Similarly, at Time 2, new landmarks e and f join the list; old landmarks a and b , which are no longer in the field of view, are dropped from the landmarks list.

Figure 3-5 presents an example of the processing for bootstrap stereo. The data set is a part of a sequence of images from the Night Vision Laboratory tape.

Figure 3-5a shows the landmark points in the first image, indicated by overlaid squares. These are the control points which we used to locate the first camera. These landmark points are then matched with their corresponding points in the second image; Fig. 3-5b shows the successful matches overlaid on the first and second images. From the image plane positions of these points, the position and orientation of the second camera are determined.


TIME
0
1
2
TRAJECTORY
a
b
c
d
e
f
KNOWN FEATURES
TERRAIN PROFILE
——— DETERMINE VEHICLE LOCATION
– – – – DETERMINE LOCATION OF UNKNOWN TERRAIN FEATURES

TIME	POSITION OF CRAFT DERIVED FROM	POSITION OF POINTS DETERMINED
0	a, b	–
1	a, b	c, d
2	c, d	e, f

Fig. 3-4 Navigation Using Bootstrap Stereo

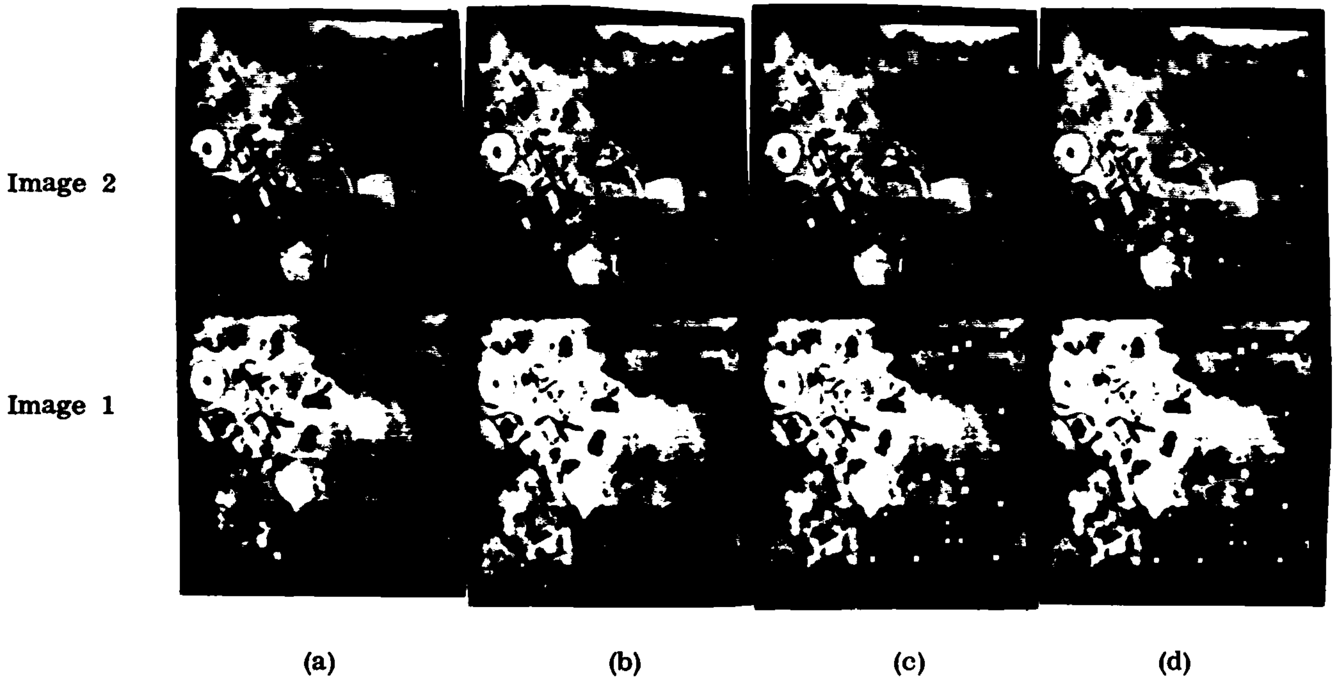


Fig. 3-5 An Example of Bootstrap Stereo Processing

(a) Known landmark points (indicated by squares) in Image 1. These are used to calibrate the camera position and orientation at Time 1.
(b) Landmark points matched from Image 1 to Image 2 (x indicates successful matches, squares are unsuccessful matches). These are used to calibrate the camera at Time 2.
(c) Landmark points (squares) in Image 1 with interesting points (asterisks) added. These are our candidates for new landmark points.
(d) Landmark and interesting points in Image 1 matched to their counterparts in Image 2 (x and + indicate successfully matched landmark and interesting points respectively; square and * are unsuccessful matches). The matched interesting points have their ground positions calculated to form new landmarks.

Next, the areas of the first image which were covered by landmarks are blocked out and interesting points are found in the uncovered areas, as seen in Fig. 3-5c. The interesting points are then matched in the second image, as shown in Fig. 3-5d. The camera calibrations for Images 1 and 2 are next used to locate the matched interesting points on the ground, forming new control points.

These steps are repeated for subsequent pairs of images in the sequences.

3.3 ERROR EXPERIMENTS, SIMULATION, AND ANALYSIS

This section covers the experimentation, demonstration, and analysis we have done on the bootstrap stereo technique. Section 3.3.1 explains the error simulations we have run and their results. Section 3.3.2 covers the demonstration tape we produced. Section 3.3.3 contains a comparison of the interest operators presented in Section 3.1.2.

3.3.1 Description of the Error Simulations

In the course of developing Bootstrap Stereo, it became obvious that we did not have a data set with known camera geometry and known ground landmarks. Despite this, it would be necessary to document the buildup of error in the camera and ground point positions as the bootstrapping progressed. The only solution was to program a means for simulating a flight, thus creating data on which the pieces of code could operate as they would for bootstrapping.

The ideal manner in which to do this would be to simulate grey-level imagery of a realistic 3-dimensional surface as seen from arbitrary viewpoints. This would by accomplished by creating a digital model of a piece of terrain – complete with features such as vegetation, roads and houses, as well as the reflectance properties of each part of the model. We would then draw up a flight path over the simulated terrain, and calculate the set of images that the simulated camera would take of its simulated world as it moved along this path. These images could then be fed directly into the interesting point programs, etc., and we could compare the resulting bootstrapped positions to the simulated flight path.

Anyone familiar with the literature of image graphics generation will recognize this as a highly ambitious task. Current programs which create images of modeled objects deal with simple polyhedral-faced objects, limited in number, and of simple colors and textures. Even so, the creation of a single frame of video requires hours of CPU time on a fairly large computer. We were talking about 50 to 100 images, with the number and texture of the objects significantly higher than that handled by most existing programs. (The interest operator and the correlation matching techniques require considerable image detail in order to work realistically.) Because we had available only an overworked minicomputer and a limited amount of time in which to program the simulation and obtain results, we had to scale down our requirements for a simulation.

We began by reexamining what we needed to simulate. The major sources of error in bootstrapping come from errors in point matching between images and the manner in which these perturb the numerical analysis and projective geometry of the camera model calculations and the ground point positioning. If we could reasonably simulate the errors in the matched image plane points, we could do away with the need for grey-level imagery.

Given this simplification, we proceeded with our simulation. We first created a digital terrain model by constructing a plane grid below the general swath of the flight path, then using a random number generator to create elevations at each point of the grid. We then devised a flight path by flying our hypothetical aircraft along a given vector, taking its position at intervals, and introducing random perturbations in the position and orientation of the aircraft (hence the camera) at each step along the way.

For each camera position, we did the necessary projective geometry to see where each of the terrain grid points fell in the image; those which fell outside of the field-of-view of the camera were discarded. Each image point was perturbed by a random amount and/or rounded (i.e., to the nearest pixel or 1/10 pixel), to simulate a match error. Image points were tagged with the ID of the grid point which generated them, so that points in two images could be matched symbolically. (Fig. 3-6 summarizes the parameters which define a simulation.)

We then proceeded to run the bootstrapping programs on this data set. The programs for locating interesting points and matching them were replaced with a single program to retrieve interesting (i.e., visible) points from the image point files and match them from file to file by their ID numbers. The camera position calculation program and the ground-point positioning program were used without changes.

This simplified simulation required about an hour of computer time to simulate the data for 50 iterations of bootstrapping and to run the programs through these data. About 25 different simulations were done, investigating the effects of various mission parameters (such as camera field-of-view, pointing angle with respect to the direction of flight, etc.) on the resulting errors. Several subsidiary programs were written to analyze and plot the error curves.

The general conclusions from these simulations are that the distance traveled before path error becomes unacceptable can be increased by:

(1) Increasing the camera field-of-view (Fig. 3-7)
(2) Increasing the platform stability (Fig. 3-8)
(3) Increasing the match accuracy (Fig. 3-9)
(4) Using backward-looking imagery (Fig. 3 -10)

The first three of these are fairly obvious. Increasing the field-of-view increases the distance which can be traveled between camera shutterings while still maintaining 75% overlap in the data. Increasing the platform stability makes it less likely that

CHARACTERISTICS		PARAMETERS
TERRAIN:	x, y Grid points, with random elevations above a level plane	o Ground grid spacing o Base plane elevation o Amplitude of elevation
FLIGHT PATH:	Straight, level course, with random perturbations in position	o Vehicle elevation o Amplitude of position perturbations
CAMERA ORIENTATION:	Fixed with respect to the vehicle, with random perturbations in vehicle attitude	o Pitch angle of camera with respect to flight path o Amplitude of attitude perturbations (heading, pitch, and roll)
SYNTHETIC IMAGERY:	Image plane projections of terrain points, with random perturbations in image plane position	o Image plane size (x, y) o Field-of-view angle o Imagery overlap o Amplitude of image plane perturbations
PROCESSING:	Normal camera modeling and ground point positioning, with symbolic image point matching	o Number of iterations performed

Fig. 3-6 Characteristics and Parameters of the Bootstrap Stereo Simulations

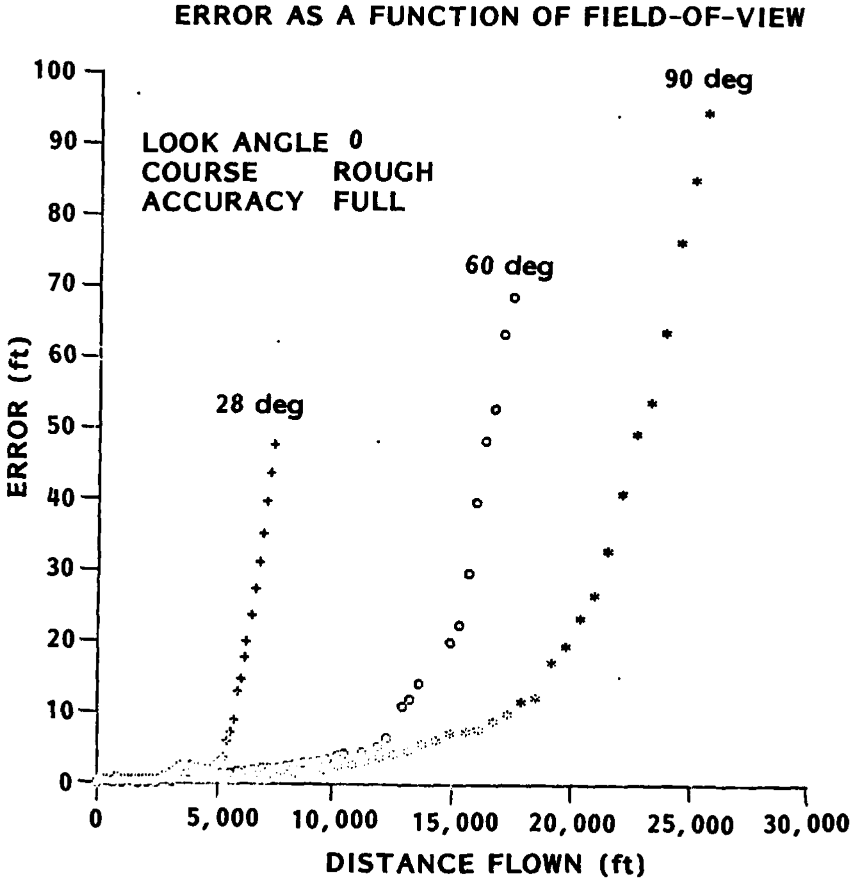


Fig. 3-7 Error as a Function of Field-of-View. Increasing the camera field-of-view increases the distance which can be flown before errors become unacceptaᵇˡe. (In this and all subsequent simulations presented here, elevation = 1200 ft, distance flown varies to maintain 75% overlap)

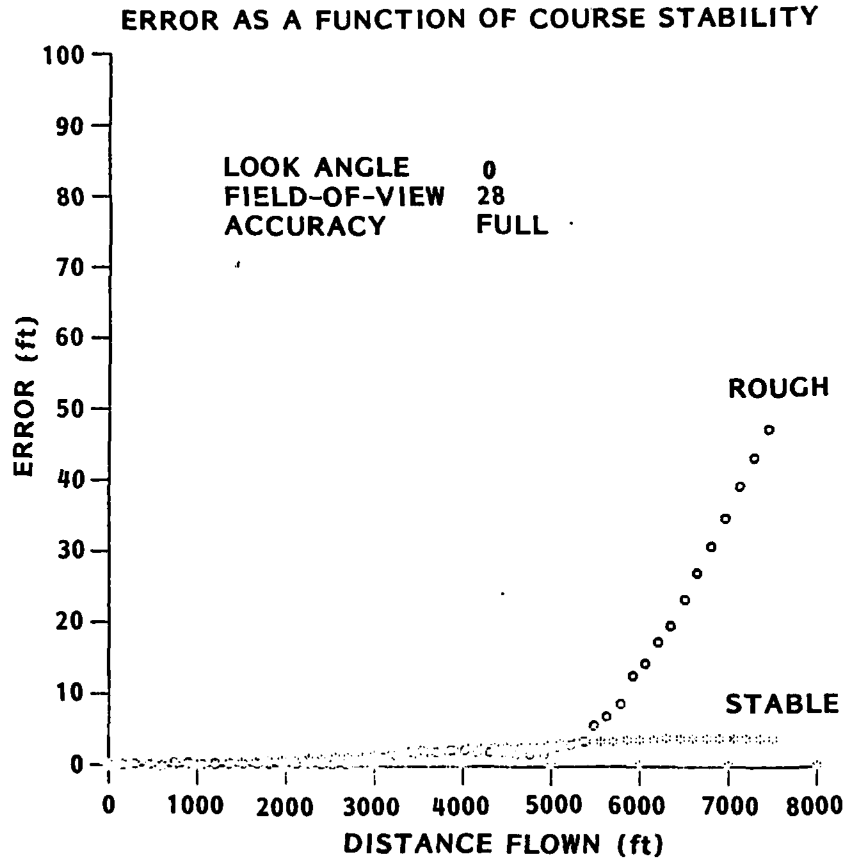


Fig. 3-8 Error as a Function of Course Stability. Increasing the platform stability decreases error buildup, especially for narrow field-of-view cameras

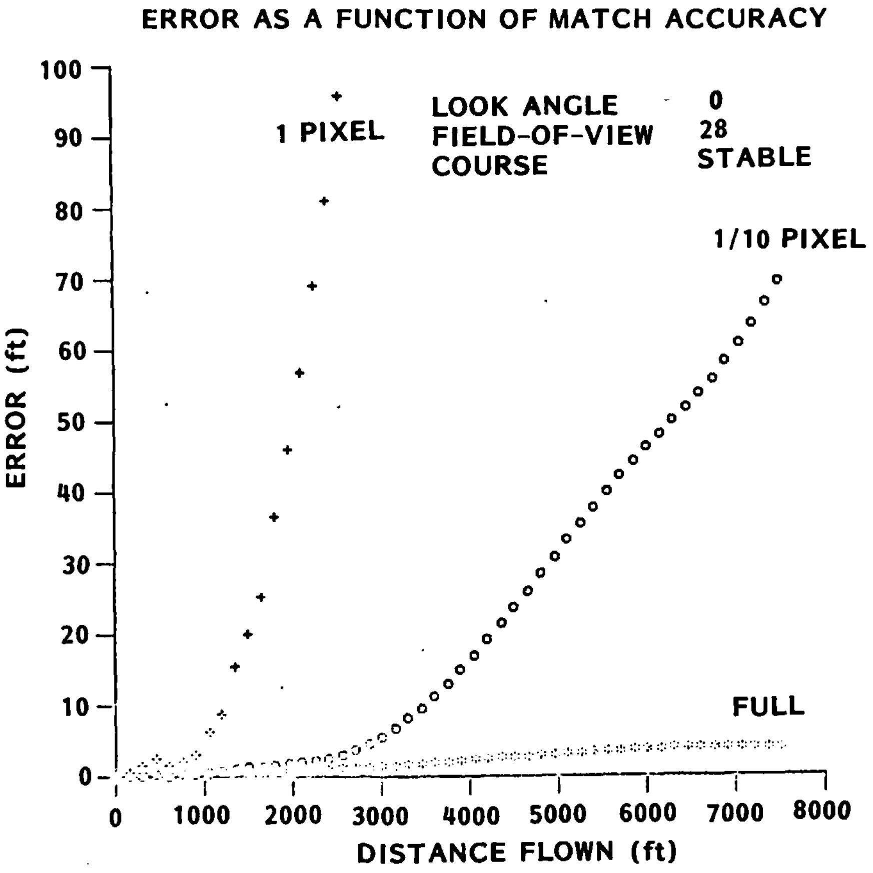


Fig. 3-9 Error as a Function of Match Accuracy. Increasing the match accuracy decreases the error buildup

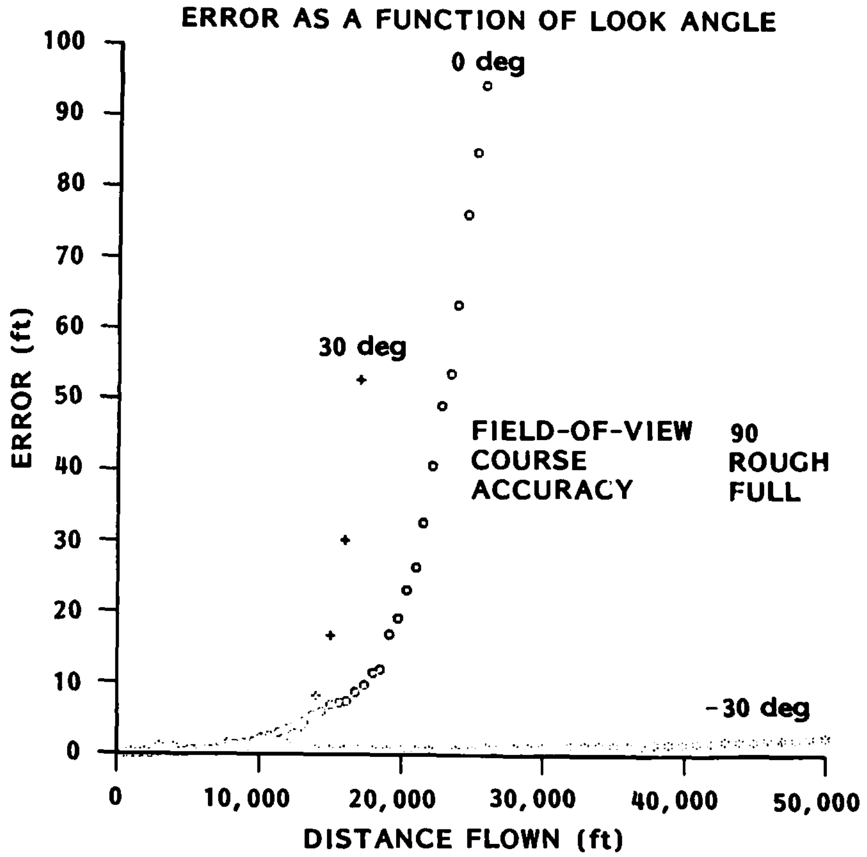
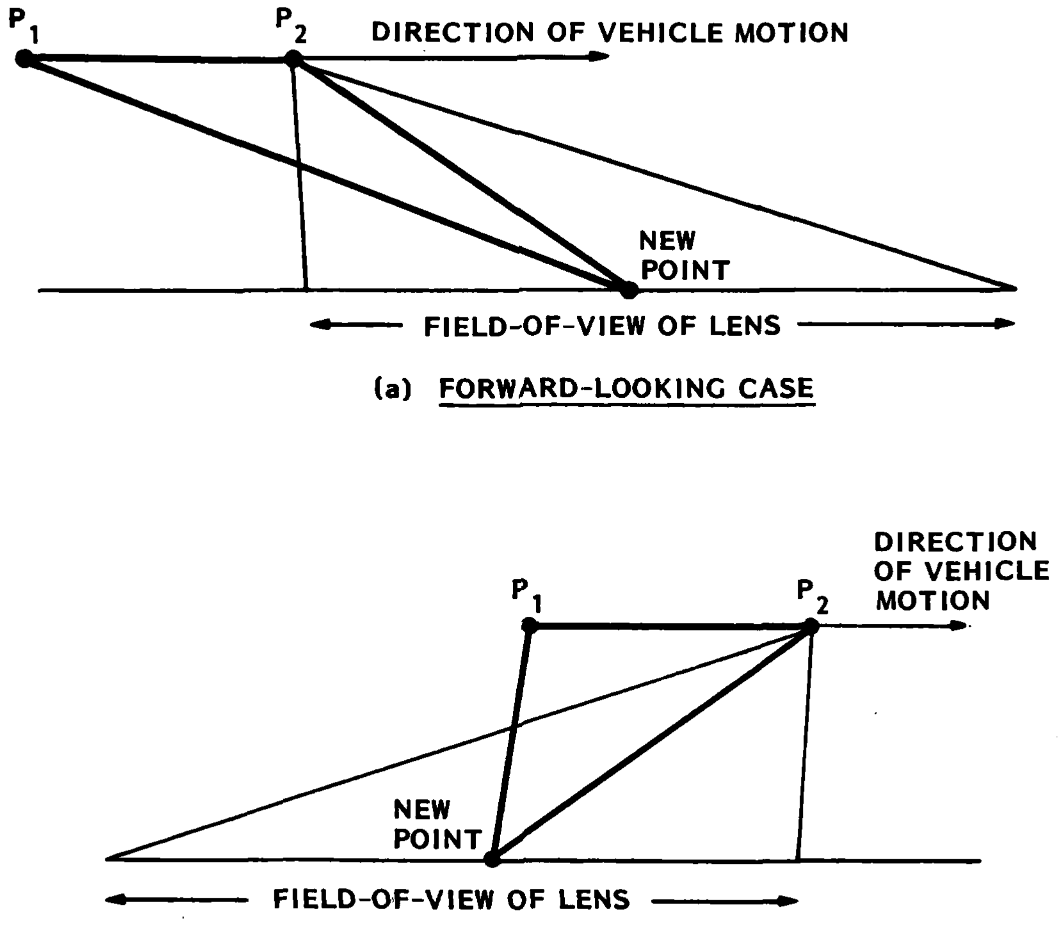


Fig. 3-10 Error as a Function of Look Angle. Using backward-looking imagery greatly increases the distance that can be flown before position errors become unacceptable

wild swings in the pointing angle of the camera will decrease the ground coverage overlap, which increases camera positioner inaccuracy. Increasing the match accuracy decreases the uncertainty about camera and ground-point positions, allowing error to build up more slowly.

That backward-looking imagery should be superior to nadir or forward-looking imagery is less intuitively obvious. To understand this, consider the two major ways in which errors enter into camera positioning. If the set of ground data points and their corresponding image points are slightly inconsistent, the camera calibrator will make an error in the camera position and orientation. On the other hand, if a pair of camera positions and of matching points within the images are slightly inconsistent, then the rays from the camera center through the image points will not intersect precisely, giving an error in that ground-point position. Of these two errors, the ground-point positioning is more sensitive, since it depends on only two rays, while the camera position determination uses information from a large number of ground-point-to-camera rays; the redundancy in the multiple observations helps greatly to reduce the error.

Now consider the geometry involved in the forward- and backward-looking cases. When the camera is looking forward, the new points are being placed far ahead of the camera by means of a very oblique triangle (see Fig. 3-11a) where a small error in match or camera orientation can cause a large error in the ground-point position. When the camera is looking backward, the new points are being placed almost directly under the camera, by means of a nearly equilateral triangle (Fig. 3-11b) – the most favorable geometry for minimizing ground-position error. Nadir imagery shares this favorable geometry, but suffers because of the small amount of visible terrain. Tipping the camera forward or backward brings more terrain into the field-of-view, permitting longer moves between images. Thus, of the three look orientations, backward-looking stereo provides the best combination of conditions to maximize the distance moved before the errors become unacceptable.



(a) FORWARD-LOOKING CASE

(b) BACKWARD-LOOKING CASE

Fig. 3-11 Effect of Forward and Backward-Looking Camera. The long, narrow triangle obtained in the forward-looking case is more sensitive to angular errors

The obvious tactical question is how far can the bootstrap technique fly before its errors become unacceptable. That, of course, will depend on the flight parameters. We ran one simulation in which all parameters were favorably set – after flying 100,000 ft (almost 20 miles), the position was off by about 25 ft, and error was still accumulating slowly. We do not know how far this flight could have gone before the errors became unacceptable, as this is the longest flight we have simulated.

It should be mentioned that most of our simulations did NOT include any use of the instrumentation on our simulated aircraft. Of course, any reasonable system which is flown will have attitude and altitude instruments whose readings at the times the camèra is shuttered will be available to the processing system. We have flown one simulation using postulated instrument readings and constraining the camera position and orientation solution to lie near these. This simulation showed a 5-fold increase in distance traveled before the position became inaccurate, when compared to a similar run without the instrumentation and constraints.

Of course, these are still only simulation results. Until we can obtain calibrated, controlled imagery with known ground truth on which to run bootstrapping, then compare its results with a simulation of the same parameters, it will be difficult to tell how accurately our simulation represents the bootstrapping process. Until that time, we are reluctant to draw conclusions about the probable distances which this technique could cover with acceptable error.

3.3.2 NVL Experiment/Demonstration

As a part of this contract, we prepared a video tape demonstrating our results in bootstrap stereo and in landmark determination. The data for the demonstration were a sequence of images from a video tape taken over the Night Vision Laboratory's terrain board. The imagery resulted from mounting a video camera on the gantry crane at NVL and having the crane's operator mov.: the camera along a generally specified "flight path" over the terrain board. The flight began over a village at the southeast corner of the model and proceeded northward over a variety of wooded, rolling terrain. Figure 3-5 shows the first two images of this flight. The original plan was to use one of these sequence of images, choose landmarks from the 12,500 scale contour map of the terrain board, determine (by hand, if necessary) the exact locations of these landmarks in an initial image, and bootstrap from there. This plan quickly ran into two snags – lack of camera calibration and lack of map accuracy.

Our camera model calculations assume that the image is formed by the projection of light rays onto a focal plane through a focal point. This is a reasonable model for standard film/lens photography, although some fine corrections are necessary for complete accuracy. Vidicon cameras, on the other hand, are notorious for their violation of this assumption – their internal geometry causes the positions of the digitized raster of image points to differ from the assumed raster, often by several pixels. Usually, this problem is handled by calibrating the distortion introduced by the vidicon (e.g., by having it view a known field, such as a grid of points), then correcting for the distortion, either by rectifying the image before processing is done or by correcting the image plane positions used within the camera model calculations.

Unfortunately, no standard calibration was available for the vidicon which made our input video tape. No measured reference was included in the area viewed by the camera during its flights. Furthermore, because of uncertainty about which camera/lens combination had been used, it was not possible to go back and obtain the calibration corresponding to our video tape.

Still, we reasoned that the terrain model is a known object. If we could select a large number of points in an image and locate them on the map of the terrain board, we should be able to derive an approximate camera calibration from them. We proceeded to try to do so.

We quickly found that the accuracy of the NVL terrain model map was not up to this task. Due to constraints imposed by the printing of maps, the National Map Accuracy Standards permit an error of up to 1/30 in. in the placement of features on a map of 1:12,500 scale. This corresponds to up to 35 ft of ground position error for each feature – entirely too much to do any accurate calibration of vidicon distortion (or even to calculate camera position, given a calibrated vidicon).

The time available in which to produce our demonstration tape did not permit us to look for new data. Instead, we decided to use the existing data to demonstrate the stereo point handoff portion of bootstrapping, with a simulation (using the approximate pointing angles, focal lengths, elevations, etc., of the input camera) to show the accuracy which could be obtained from such a flight.

In theory, this point-handoff demonstration could have been done using just the interesting point and the point-matching software. It is known, however, that the point-matching program occasionally makes mistakes, which in a real system would be caught by the camera model solving program or the ground-point positioning program. To add some of their point-editing capability to the demonstration, we used a different camera calibration package (Ref. 7) to solve for the relative angles between the two cameras, and let it remove undesirable points.

The flight parameters used in creating the input data tape were far from optimal from the point of view of maximizing bootstrapping accuracy – NVL used a narrow-angle camera, pointed forward along the flight path. It was not surprising that a simulation using these parameters and a reasonable image plane accuracy (1/10 pixel) showed that the vehicle would be hopelessly lost within a mile of the beginning of the flight.

Any reasonable system would have onboard vehicle attitude sensors, which would tell the bootstrapping system the orientation of the vehicle (hence of the camera) at the time each image is taken. No such sensor data were available for the NVL input data, but we did know that the flight was approximately straight and level, so that the attitude at each point was approximately constant. We added to the simulation the capability of using constant attitude information, and constrained the camera model solution to lie near this attitude. In this mode, we were able to get 4 miles of flight before the solution became unreliable. This was the simulation presented in the video demonstration.

3.3.3 Interest Operator Evaluation

As mentioned previously, there are three different interest measures which we have used. Standard deviation needs no explanation, being a well-defined statistical quantity. The idea behind the use of standard deviation as an interest operator is that it has a low value in areas of low information, where correlation performs poorly as a means of matching images.

Directed variance is somewhat misnamed; it actually is minimum RMS directed difference, but the name that Moravec (Ref. 9) used in his papers has persisted. The idea is to calculate the difference between adjoining pixel intensities in four directions. [If the pixels are labelled

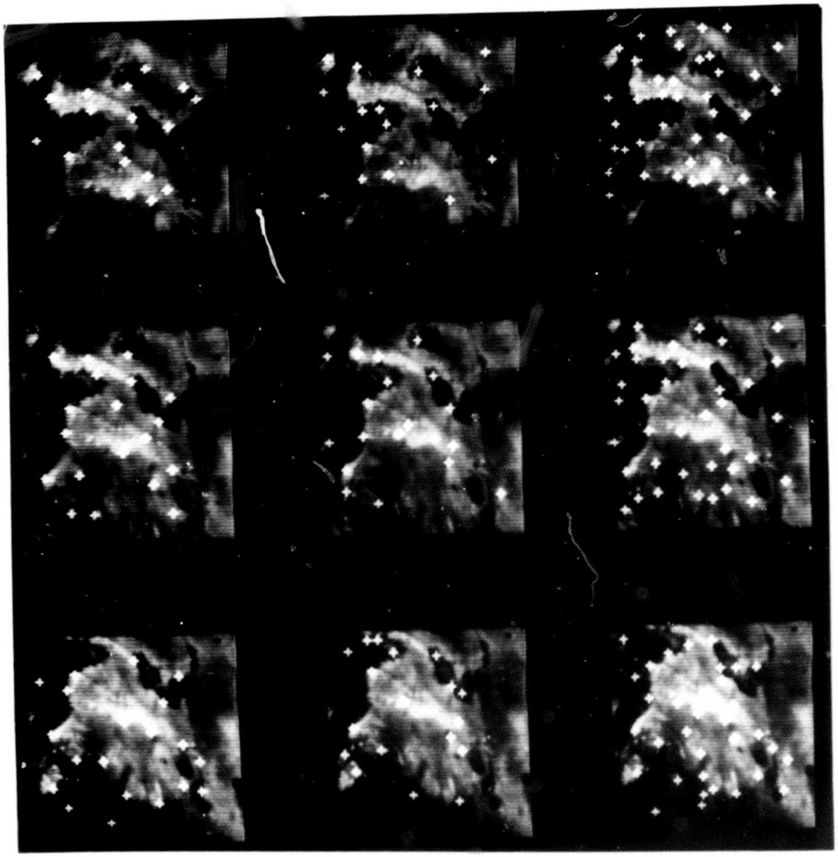
A B
C D

then the four differences are (A - B), (A - C), (A - D), and (B - C).] These differences are each squared and summed over a window. (In the definition of the measure, the square root is taken, but since we are interested in relative maxima, we do not take the square root on any of these measures; this expedites the computation.) If the area has no information, all of these "directed variances" will be small; if the area has a strong linear edge with little information on either side, then the directed variance which most nearly parallels the edge will be small. By defining the interest measure to be the minimum of the directed variances at a point, we can reject not only areas with low information but also areas with one-directional information, which can also defeat correlation matching.

Edged variance is a combination of these two measures. It first forms the directed variances as above but then uses their ratios in perpendicular pairs, that is, S(AB)/S(AC), S(AC)/S(AB), S(AD)/S(BC), and S(BC)/S(AD). The minimum of these ratios gives a measure of the relative strength of the information parallel to the dominant edge in the window, independent of the contrast of the image. Since it does not give any indication of the amount of information in the window, we have multiplied it by the standard deviation. This combined measure seems to give a better scattering of locally interesting points than does either of its two components.

Figure 3-12 shows the application of these three interest measures for a sequence of three images taken over the Night Vision Lab terrain model. The images in the first column have overlays showing the peaks in ordinary variance; the second column shows peaks in directed variance; the third column shows peaks in edged variance. Note that ordinary and directed variance find points along the strong, irregular edges which mark the tree/grass boundaries, and ignore open areas having more subtle features. Edged variance, on the other hand, gives a combination of strong and subtle features, while avoiding excessively plain areas.

On the basis of performance, edged variance is the preferred interest measure. It is, however, the most expensive of the measures, requiring the calculation of six sums (five of which are squared quantities), four divides, a MINIMUM operation, plus two multiplies and a divide to form the standard deviation. Directed variance is less expensive, requiring four sums (all of squared quantities) and a MINIMUM. Standard deviation is the cheapest, requiring two sums (only one is a square), plus the two multiplies and a divide.



The first column of images shows peaks in ordinary variance for three images from Night Vision Laboratory's terrain model. The second column shows peaks in directed variance for the same three images. The third column shows peaks in edged variance.

Fig. 3-12 Three Measures of the "Interest" of a Point

If the scenes to be processed contain mostly natural terrain, the edge rejection properties of directed variance and edged variance are not needed, and the cheaper standard deviation should be used as the interest measure. If the scenes contain many linear features, such as roads, then edge rejection is important, and one of the other two measures must be used. In relatively featureless terrain, the increased performance of edged variance is needed; in terrain which will give fairly contrasty images, the cheaper directed variance can be used.

3.4 MECHANIZATION AND SPEEDUP

The emphasis to date on the Bootstrap Stereo package has been on implementing an effective means for accomplishing the navigational objective of using visual information to keep track of the vehicle's position in the real world. The result has been an experimental package which accomplishes the objective without addressing operational realities such as timing and size constraints.

The current implementation of bootstrap stereo is as a set of separate but cooperating programs on a Data General Eclipse computer. Table 3-1 shows the sizes and timings of these programs when operating on a pair of 256 x 256 images. The "bottom line" is that a single iteration of bootstrapping by the present system takes on the order of 5 minutes, and uses 120 kbytes of program storage plus 256 kbytes of image memory.

A vehicle flying at 1200-ft elevation and using a 90 deg field-of-view lens pointed backward 30° (the most accurate angular configuration found to date) can view a swath of ground 4800-ft long. To maintain 75 percent image overlap, such a vehicle can move 1200-ft between images. At 150 mph, this distance is covered in about 5-1/2 s, therefore this would be the desired basic cycle time for an iteration of bootstrapping in this configuration. This requires a speedup factor of 50 for the present bootstrapping algorithm to function in this environment.

Table 3-1 SIZE AND TIMING INFORMATION FOR BOOTSTRAP STEREO, AS CURRENTLY IMPLEMENTED.

	Size (Bytes)	Overhead Time (Seconds)	Run Time (Seconds)
Interesting Point Selection	24,067	5	40
Point Matching	34,816	5	(132)
Matching Through Reduction Hierarchy (Low Levels)			19
Matching Through Reduction Hierarchy (Top Level)			59
Matching Through Landmark Points			21
Matching Through Interesting Points			33
Camera Model Calculation (Time Depends on Number of Editing Iterations)	18,432	5	8 - 32
Ground Point Position Calculation	15,360	5	3
Data File Cleanup for Iteration	9,728	5	6
Image Data Preparation	19,968	5	58
Total	122,371	30	247 & up
	120 kbytes	277 sec	4-2/3 min

Let us examine where this speedup could come from. The present implementation is segmented into small pieces, both for developmental flexibility and because the computer on which it was written restricted code size to 42 kbytes. Consequently, each program must be loaded separately, and point data must be communicated between programs on disk files. Image data are transferred from disk files to a separate image memory by the image data preparation program; other programs read these data from the image memory into computer memory in small pieces, as needed. All of this takes time; the package would run considerably faster if all the programs and data could reside in a single, suitably large computer memory.

Further speedups would occur with the removal of the diagnostic printouts and visual cverlays that currently allow the user to monitor program progress. Overall, this type of cleanup could result in a speedup factor of 2.

The timings also reflect the fact that the implementation is a suboptimal coding of breadboard algorithms in FORTRAN on a not particularly fast computer. The actual deployment of these algorithms would be as pieces of hardware after the algorithms have been streamlined and optimized. Conservative estimates place the speedup factor of going to a hardware implementation at 50 to 100 for generalized image processing functions; speedups of 10,000 have been demonstrated in some cases.

Considerable speedup will also result from the actual streamlining of the algorithms. An example is the incorporation of velocity information into the finding of corresponding points. If one knows the ground velocity, one knows approximately where points from one frame will appear in the next frame, since one has an estimate of where certain known points lie on the ground, and can make an estimate of the new vehicle position, given the velocity information. The ground points could then be projected back into the image plane at the new location of the camera. The actual corresponding points in the image plane could then be found by a spiral search centered on the approximate projected location. This should speed the correspondence computation (a significant part of the overall stereo computation). The present algorithm suffers also because of its sequential implementation. The computer programs which make up bootstrapping currently run separately, one at a time. The processing flow, however, permits some of these functions to be performed simultaneously, as shown in Fig. 3-13. This permits another speedup of a factor of 2. Within programs further parallel processing is possible, since most of the programs perform the same calculations on a set of points. Obviously, 16-way parallelism within each program would give a speedup factor of 16.

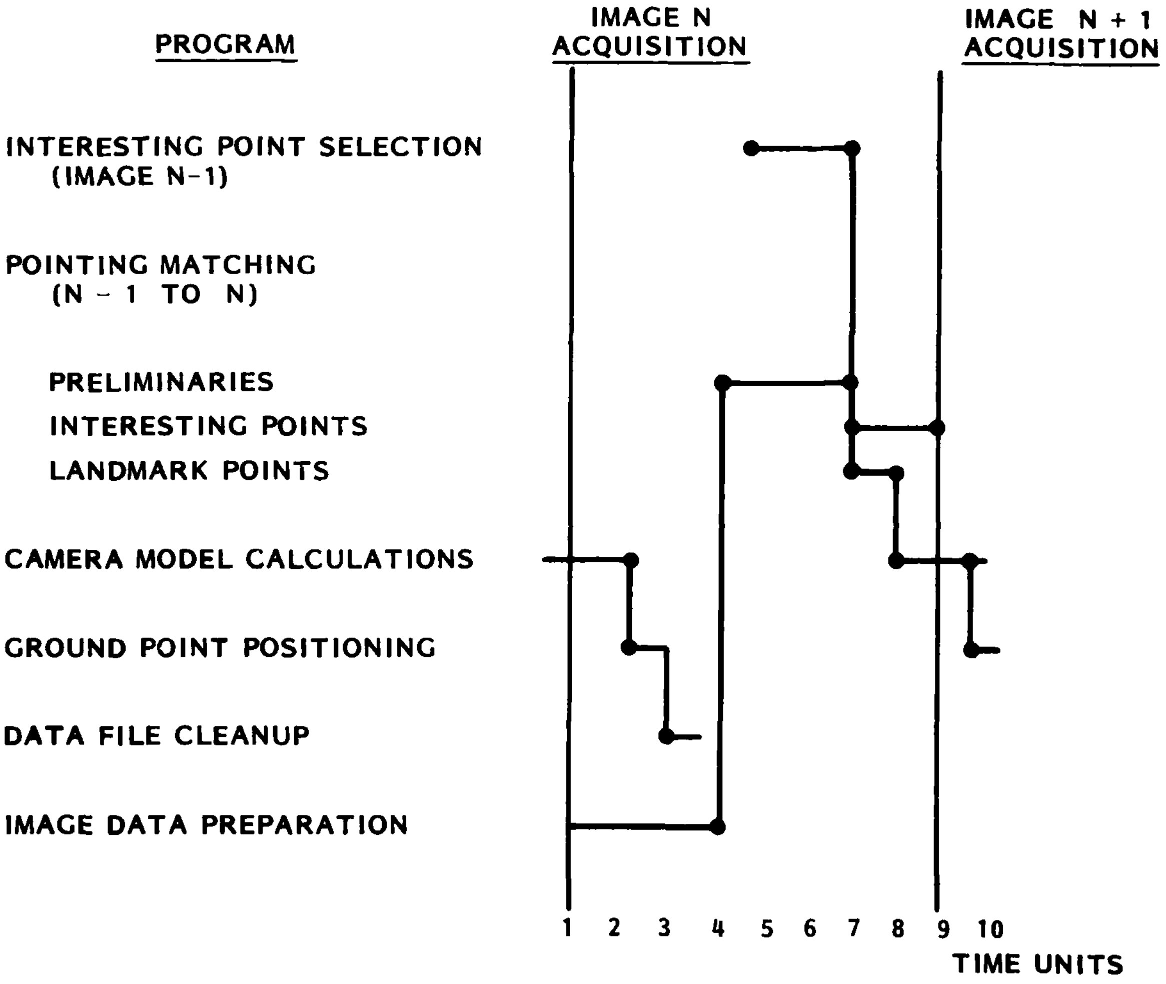


Fig. 3-13 Overlapping of Stereo Processing to Obtain Speedup

Overall, the combination of refining the algorithms, mechanizing them, and applying parallelism where applicable could result in speedups ranging from 100 to 10,000, depending on the precise manner in which it is done. Since only a speedup of 50 is needed to make the current technique feasible, it is clear that some form of bootstrapping could be usable as a navigation aid.

Section 4
LANDMARK SUBSYSTEM

The systems aspects of the Landmark Subsystem were described in some detail in Section 2. The present section describes the landmark analysis work using intensity images as the source of landmark information. Other alternatives are stereo-derived topographic landmarks and actively sensed range images.

4.1 LANDMARK DETERMINATION USING INTENSITY IMAGES

In many image identification environments, edges (locations of high intensity gradients) characterize significant attributes of a scene. However, initial processing by edge operators provides very local (pixel-level) information about these points of high contrast. This local information is not in a form amenable for identification or matching of the major constructs in the image. There have been many approaches described in the literature which attempt to form intermediate or higher level data structures which can then serve as more reasonable primitives for the identification or matching process. For example, Nevatia and Babu (Ref. 12) link edge elements based on proximity and orientation and then approximate the linked elements by piecewise linear segments. Relaxation has also been applied (Ref. 13), using iteration to refine the probabilistic interpretation of individual pixels as edges based on their neighbor's interpretations. Many of these approaches share a common defect; the decision to join edges or bridge gaps is made on the basis of local information within the gap. Our approach joins edges and links across gaps only if the two candidate points appear to bound the same region. Thus edge points in the vicinity of a corner (whose directions differ by 90° or more) are still associated as part of the same boundary. Using this idea, it is possible to filter the linking process and extract robust boundaries (strands) of significant length. The rest of this section describes our investigations as to the utility of using these strands for locating in actual imagery landmarks derived from maps, which then can be used for position update.

4.2 CONTOUR DETERMINATION USING SYMCHAINS

Low-level edge processes are frequently used to identify discontinuities. Because of noise, shadows, etc., in real-life images, these local edges do not always fit into extended, nonconflicting boundaries. One type of tracking approach (Ref. 14) ranks the edge sequences formed by the low-level process using a confidence measure and then considers edge sequence associations in an ordered fashion to determine whether separated edge sequences can be combined into a more extended boundary. Another set of techniques, e.g. the Hough transform (Ref. 2), transforms the edges into an alternate domain and associates clusters in the transformed space with long boundaries.

Our "symchain" approach consists of the following steps:

(1) Image elements that are the edges of objects are identified

(2) Boundaries are tracked, resulting in a list of edge pixels and their neighbors on the boundary

(3) The best left and right neighbors for each pixel are then selected as link elements

(4) The resulting connected set of links are formed into symchains

(5) Symchains are represented symbolically in terms of link element length and angle which are called strands

(6) This description is then compared with reference strand descriptions in the landmark data base

The approach assumes that a low-level edge process has been used to determine candidate edges (Ref. 15). Let E be the set of edge points and let $e \in E$. Suppose at some threshold, t, there is a connected component of points at or above threshold whose boundary includes e. (We call such boundaries "contours.) Let $e = e_1, e_2, \ldots, e_n, e_{n+1} = e$ be the succession of edge points encountered in a clockwise traversal of the contour. We define $C(e,t) = e_2$ $(CC(e,t) = e_n)$ as the clockwise (counterclockwise) neighbor of e. Each neighbor $C(e,t)$ delimits a path from e to $C(e,t)$ along a contour. At a different threshold t', $C(e,t')$ might delimit a different path (Fig. 4-1). If no contour passes through

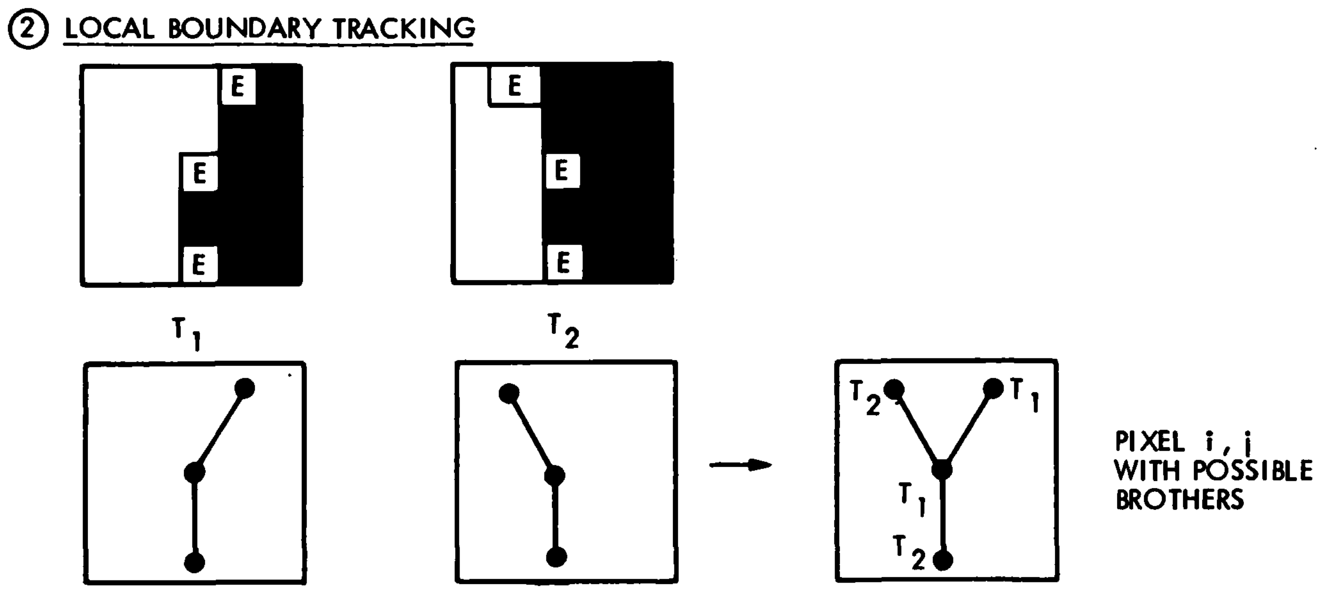


Fig. 4-1 Different Edge Contours are Identified at Varying Thresholds

e at a given threshold t or if exactly one other edge point lies along the contour (i.e., $C(e,t) = CC(e,t)$), then $C(e,t)$ and $CC(e,t)$ are left undefined.

Consider the collection (including duplicates) of clockwise neighbors $C(e,t_1)$, $\ldots, C(e,t_k)$ for some set of thresholds $T = \{t_1, \ldots, t_k\}$. In general, some neighbor e' will be the clockwise associate occurring most often for the various thresholds T. Define C(e) to be the clockwise neighbor of e. The counterclockwise associate CC(e) is defined similarly.

When completed, the process has selected for each edge-point e (at most) one clockwise associate and (at most) one counterclockwise associate and has computed their figures of merit. Note, however, that the association is not necessarily mutual (symmetric), i.e., it is not true that $CC(C(e)) = e$. This is reasonable since it is possible for an edge-point to be in the vicinity of a corner at which three or more surfaces meet. It may also result from the breaking of ties. Nonetheless, for images in which the edge extraction process has produced thin (1-pixel wide) edges, the great majority of linkings turn out to be mutual, providing additional evidence of their correctness. Such links are called "symlinks."

The symlinks can be aggregated into sequences of maximal length called symchains (Ref. 16). The symchain associations can be extended to include as sequences chains for which the associate of an edge is part of another symchain (Fig. 4-2), although the pixel's association is not mutual. This permits alternative strongly supported chains to be included as competing interpretations. Only the resulting symchains exceeding a minimum length (currently, eight) are retained. A subsequent test determines whether edges of these long symchains exist within prespecified distances, and, if so, they are also combined into single entities (Fig. 4-3). These are called strands. The strands are polygonally approximated using a split-and-merge technique (Ref. 17). These polygonal approximations become the basic match components (Ref. 18). The basic steps in this process described above are illustrated in Fig. 4-4.

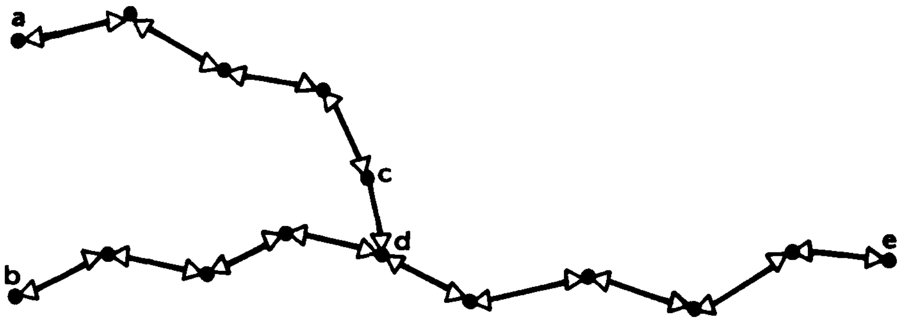


Fig. 4-2 Sequence of Symlinks from b to e form a Symchain. Sequence from a to e is also considered a symchain although link from c to d is not symmetric

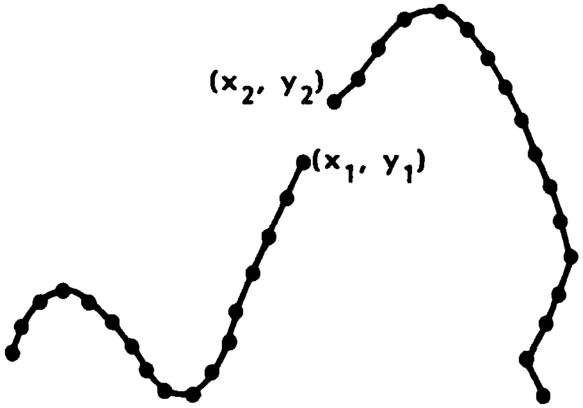


Fig. 4-3 Symchains a and b combined if the distance between (x_1,y_1) (x_2,y_2) is small enough

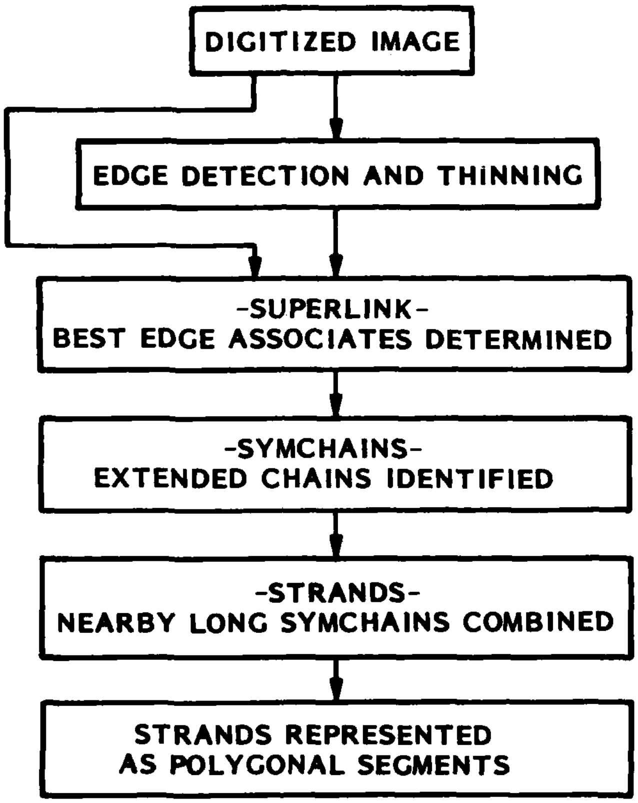

DIGITIZED IMAGE
EDGE DETECTION AND THINNING
-SUPERLINK-
BEST EDGE ASSOCIATES DETERMINED
-SYMCHAINS-
EXTENDED CHAINS IDENTIFIED
-STRANDS-
NEARBY LONG SYMCHAINS COMBINED
STRANDS REPRESENTED
AS POLYGONAL SEGMENTS

Fig. 4-4 Steps in Processing of Edge Contours

4.3 SYMBOLIC MATCHING

Local Matching

The matching procedure uses as input the polygonally approximated strands determined from the two images being compared. The following method is used. Each polygonal boundary segment is represented as a sequence of lengths and angle values (Fig. 4-5). A match is found between two boundary segments b_1 and b_2 whenever both of the following conditions are satisified:

(1) b_1 and b_2 are of the same type i (either length or angle)

(2) the difference (b_1,b_2) is less than VAL_i, where VAL_i is based on type (for the example case, where b_1 and b_2 are angles, b_1 and b_2 were considered a match if $|b_1 - b_2| \leq 30°$)

A maximum match value is computed for each pair of strands as the maximum number of consecutive matches between two polygonal boundary segments. This is called the match length. Note that the match constraints allow a fair degree of size and angle variability. This maximizes the ability of the algorithm to obtain valid matches even in the presence of scale, rotation, and perspective effects.

Global Matching

The output of the local match algorithm is a set of reference/sensed pairs of strands which have been associated. Although identifying many local matches may imply that there is a correspondence between reference and sensed scenes, a more convincing test is whether some subset of the local matches can be combined to form a consistent set of global solutions, since this can provide stronger evidence that the sensed image is viewing the same scene as the reference. Briefly, the global match program checks pairs of reference/sensed local matches, associates them if they are globally consistent, and groups them into consistent entities. The group containing the largest number of consistent matches, weighted by the match lengths of each match pair, is chosen to be the global match. However, if this group does not contain matches from differing parts of the images, it is not accepted as a viable match.

The algorithm which identifies consistent global match pairs is detailed below.

Consider two strings in each of the reference and sensed scenes. Call the reference strings R_1 and R_2 and the corresponding sensed strings S_1 and S_2. For each string, designate the starting location as (x_1^T, y_1^T) and its ending location as (x_L^T, y_L^T) where $T \in \{R_1, R_2, S_1, S_2\}$. In order for the 4 strings in the two images to be a global match, positional relationships between the pairs of strings must be similar. The distance differential criteria used are:

$$x_1^{S_1} - x_1^{R_1} \simeq x_1^{S_2} - x_1^{R_2}$$

$$x_1^{S_1} - x_L^{R_1} \simeq x_1^{S_2} - x_L^{R_2}$$

$$x_L^{S_1} - x_1^{R_1} \simeq x_L^{S_2} - x_1^{R_2}$$

$$x_L^{S_1} - x_L^{R_1} \simeq x_L^{S_2} - x_L^{R_2}$$

and similarly for the y's.

$A \simeq B$ if either $|A - B| \leq 5$ or $3* \mathrm{MAX}(A,B) \leq 4 * \mathrm{MIN}(A,B)$.

In addition, angle criteria must be met, e.g.,

$$\tan^{-1}\left(\frac{x_1^{S_1} - x_1^{R_1}}{y_1^{S_1} - y_1^{R_1}}\right) \simeq \tan^{-1}\left(\frac{x_1^{S_2} - x_1^{R_2}}{y_1^{S_2} - y_1^{R_2}}\right) \text{ etc.}$$

where $\tan^{-1}(A) \simeq \tan^{-1}(B)$ if $\left|\tan^{-1}(A) - \tan^{-1}(B)\right| \leq 45°$.

This match algorithm can be extended to include interior points of the symchains to aid in global matching.

4.4 EXPERIMENTAL RESULTS

The NVL terrain model was used as a source of imagery. Three studies were made. They were:

- o Determine whether the extracted strands represent significant features of an image
- o Determine whether strands similar to ones located in sensed imagery can be obtained from a map representation
- o Determine whether the existence of global matches can be associated with the determination of a match in a sequence of images

(1) Strands as Features

Figures 4-5 and 4-6 illustrate some of the steps which were performed on the original image to obtain the polygonal representation (strands) of the significant edge features. The a section of both figures shows the original digitized image. The b sections show edge pixels after nonmaximal suppression and thinning. The c sections illustrate the best clockwise and counterclockwise links that were determined for each edge. Intensities are proportional to the confidence placed in each association. Section d shows symchains of length 8 or greater. Section e illustrates the polygonal approximations of the strands. Note that many significant edge features are captured.

(2) Association with Map Features

In the second test, the outline of a lake was traced from the topographic map of the artificial terrain model. This was then digitized and is illustrated in Fig.

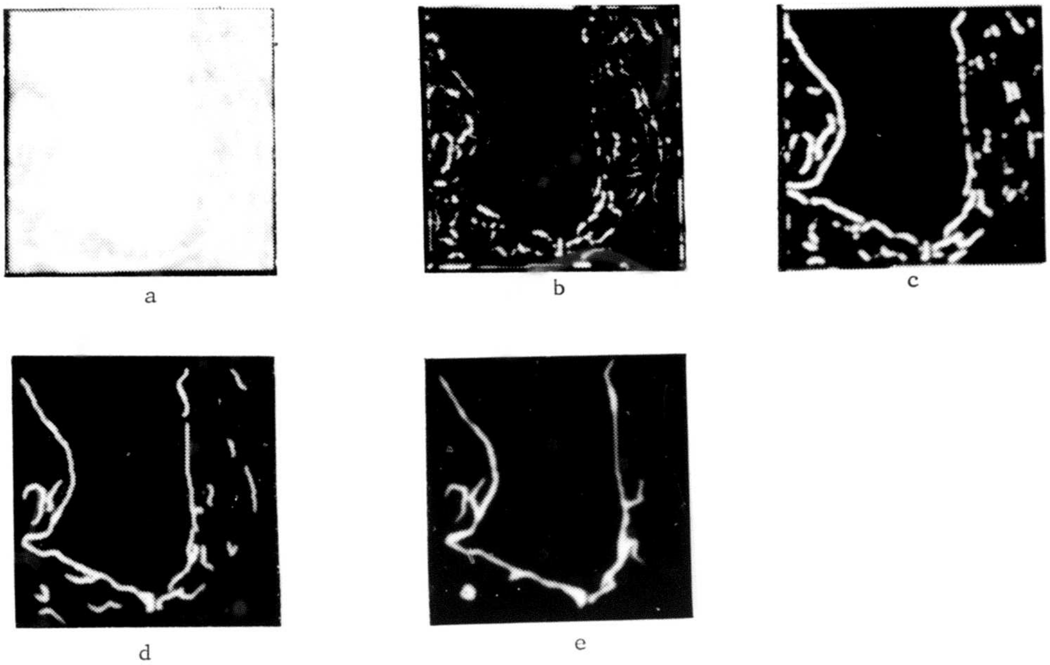


Fig. 4-5 Steps in Processing Lake Image

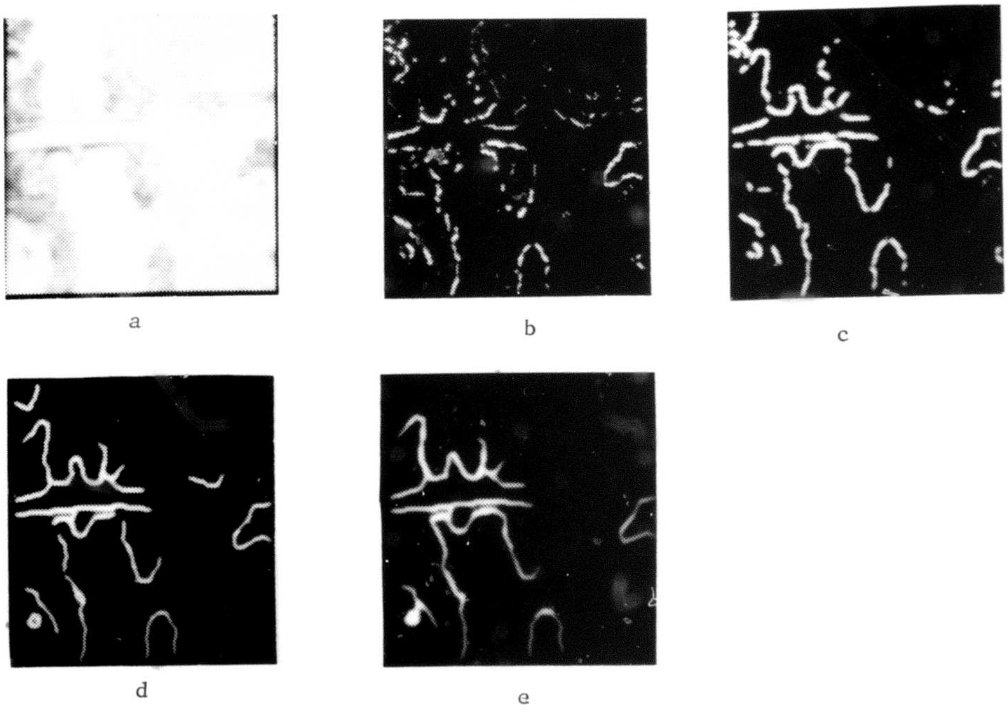


Fig. 4-6 Steps in Processing Bridge Image

4-7a. (The terrain map itself was not digitized, as it contained a confusion of ancillary markings.) This image was processed in the same manner as the actual terrain imagery. The polygonally approximated strands of the lake map are shown in Fig. 4-7c. These strands were compared to those which had been obtained from the sensed imagery (Fig. 4-7d). Match lengths of value 4 or greater between the polygonal strands are shown in Fig. 4-7e and f. Note that significant portions of both images seem to be matched even though size and perspective differences between the images are substantial. The lake image was also compared to a sensed image taken from a different view. The results are shown in Fig. 4-8. Note that there are match segments of significant extent. These matched strands, after consistency checking, can then be used as input to a procedure which provides position error information.

(3) Global Matching for Landmark Identification

A sequence of four images of the NVL terrain model illustrates the steps in matching (Fig. 4-9). Images 33 and 34 contain a bridge whose reference representation appears in the landmark database; images 32 and 35 do not contain the bridge. For each image, the reference strands for the bridge are shown on the left and the strands extracted from the image appears to the right. The images labeled a show the original strands for the reference and sensed image, b images show the strands that matched locally, and the c images show the best match satisfying positional relationships. It will be noted that only in the bridge images were global matches (consistent matches in different parts of each image) obtained. Note also that global matches were obtained in both sensed images even though the forward look angle at which the bridge was seen in the image ranged from 50° in Image 33 to about 30° in Image 34.

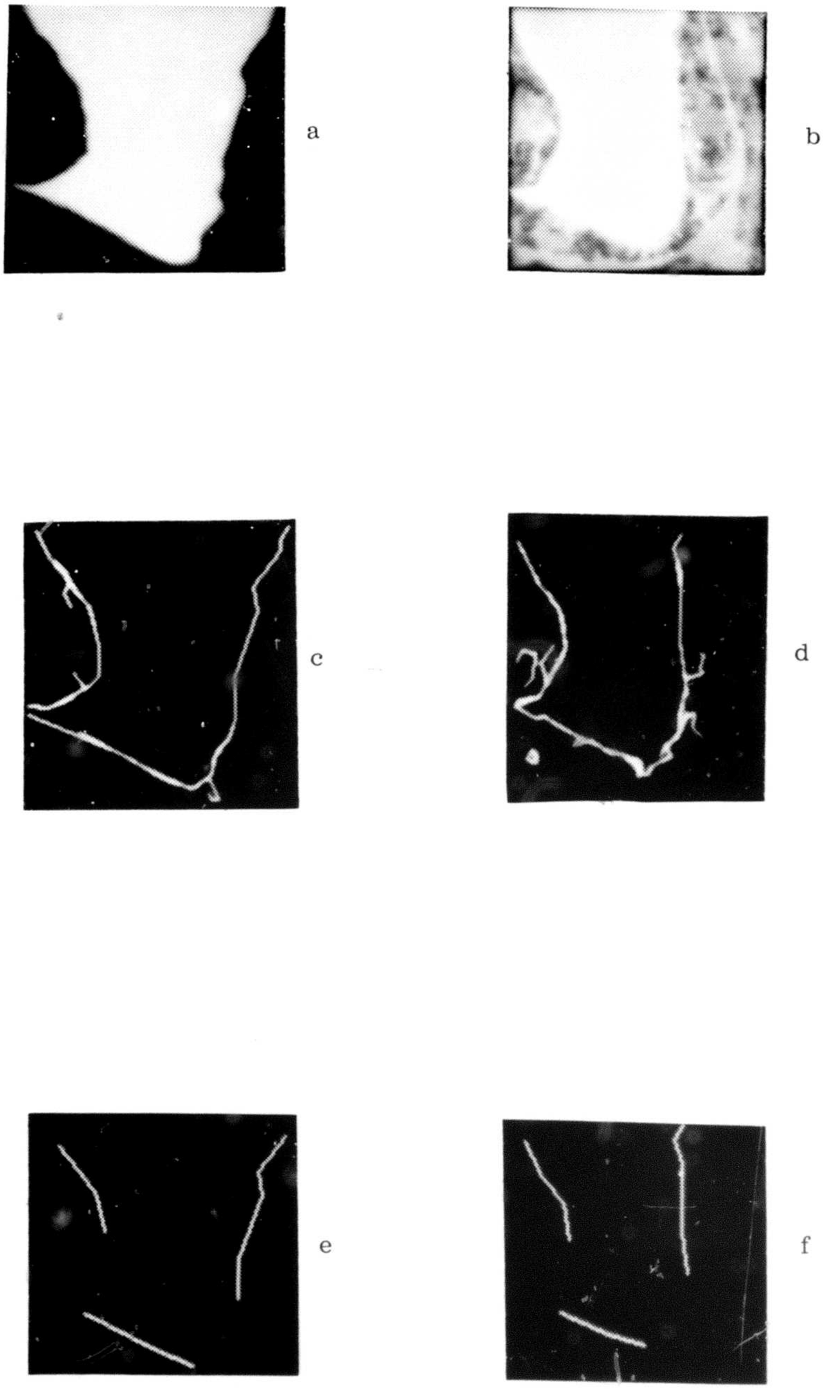


Fig. 4-7 Matching to a Reference Derived from a Terrain Map

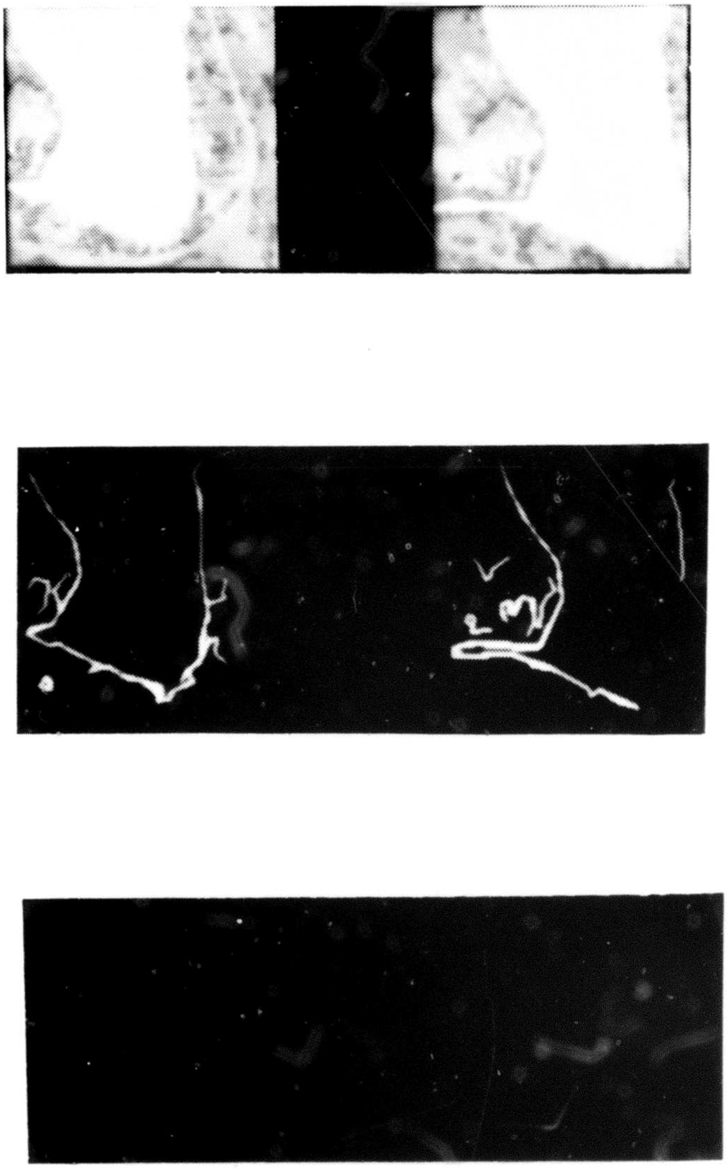


Fig. 4-8 Matching of Two Sensed Lake Images from Different Views

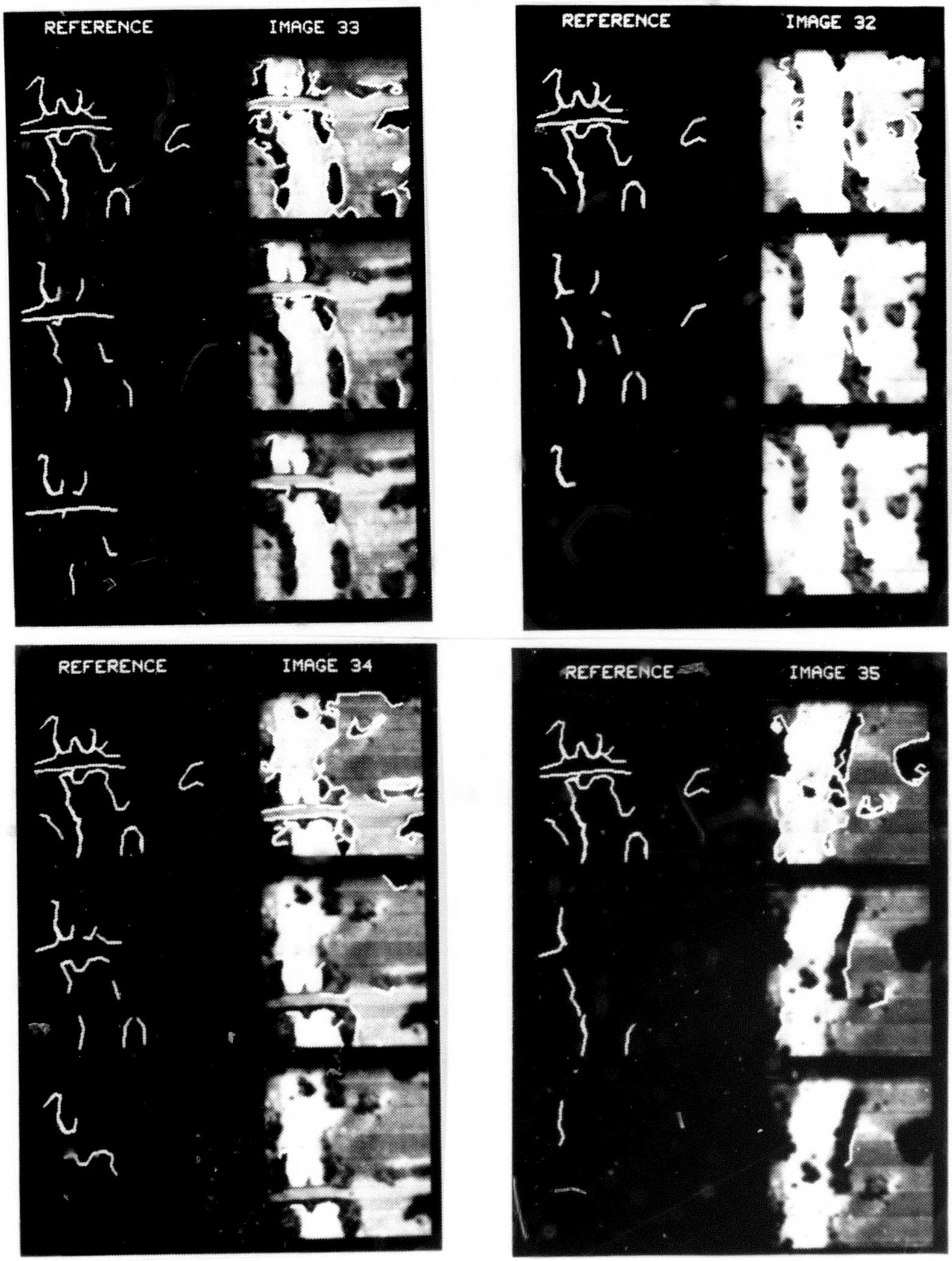


Fig. 4-9 Matching of Reference and Sensed Symchains for Bridge Reference

Section 5
SUMMARY AND DISCUSSION

This section provides a brief summary of the efforts for each subsystem and discusses important aspects of each.

5.1 SYSTEM ASPECTS

The effort in the overall system has been mainly conceptual, with the interaction of the various subsystems defined, as given in Section 2. Of particular interest is the redundancy in vehicle location obtained from both the stereo and the dead-reckoning subsystems. Both subsystems provide a confidence measure for the location estimate, with the stereo estimate based on computation residuals and the dead reckoning estimate based on the "age" of the wind velocity estimate. Actual integration of the subsystems will not be possible until the subsystem programs are converted from the present experimental form to a more efficient and modular form.

5.2 STEREO SUBSYSTEM

The stereo system described in Section 3 is the focus of the navigation system, since it not only performs the "bootstrap navigation," but also can obtain the ground velocity and the relative altitude of the vehicle. Unlike our original Image Velocity Sensor (IVS) concept, the stereo subsystem has the advantage of being independent of aircraft orientation, since its camera model solver can deal with a camera that is not aimed at the nadir point. During the course of this study, the automatic handoff of points from frame to frame, required in the bootstrap concept, was completed and demonstrated. Because calibrated imagery was not available, the error demonstrations were limited to simulations using synthetic data. From the simulations, we determined the best combinations of look angle and field-of-view so as to maximize the total distance that can be bootstrapped before a landmark fix is required.

The bootstrapping process is far from infallible. The errors to which it is vulnerable fall into roughly four categories:

(1) Loss of overlapped imagery
(2) Errors in matching control points
(3) Errors in camera calibration
(4) Errors in control-point positioning

The bootstrapping process could lose its needed overlap in imagery if the terrain over which the vehicle is flying is obscured by clouds. If the terrain is essentially featureless (for example, a large body of water or a desert), then the bootstrapper will be unable to find and match sufficient control points to continue. Similarly, several dropped frames resulting from a temporary equipment failure could cause the bootstrapping process to abort.

A human navigator faced with clouds or featureless terrain would simply shift "mental gears" and fly on instruments and dead reckoning until more favorable terrain was found. He would then attempt to locate new landmarks from which to reorient himself. Mimicking this, when stereo bootstrapping loses its overlapped imagery, the Stereo Subsystem reports failure to the Navigation Expert, which then proceeds to rely on its Dead Reckoning Subsystem until its Landmark Subsystem can recognize some new landmarks from which to reorient the Stereo Subsystem for bootstrapping. Until then, the Stereo Subsystem processes any available image sequences to extract ground velocity for the Dead Reckoning Subsystem.

The other three problems (2), (3), and (4) rarely cause the bootstrapping process to fail completely. Instead, they interact to create errors in the vehicle positions determined by bootstrapping. Since these errors are somewhat inevitable, it is anticipated that the Landmark Subsystem will be invoked periodically to search for checkpoint landmarks. Course corrections will then be determined from these checkpoints, and the bootstrapper will be reinitialized.

Gross errors in match, which can occur due to repetitive textures or moving objects, are likely to be caught by the autocorrelation thresholding or by the depth consistency or camera model consistency requirements. Small errors in match, such as would result from improper subpixel registration of the data, can slip through; these will bias the camera calibrations and control point locations slightly.

Errors in camera calibrations result either from errors in the data or from insufficient precision in the calibration calculations. The latter can be designed out of the system by ensuring that the processor has sufficient word-length and/or floating-point precision to handle matrix inversion. Errors in the data are most likely to affect the camera position, as its orientation is fairly well known from the vehicle orientation reported by the Instrument Subsystem. Techniques exist (Ref. 6) for the adjustment of the image data points along with the camera parameters, to produce more consistent results.

Errors in the positions of the original landmarks will be propagated through the bootstrapping chain. Such errors should be static, however, and should only result in small perturbation in the vehicle location. Errors in control point positioning are interrelated with errors in the match point location and the camera calibration. Using the redundancy inherent in multiple images (Ref. 6) can resolve some of these uncertainties.

5.3 LANDMARK SUBSYSTEM

In our system, a landmark identification is required:

- When the bootstrapper starts to diverge, say every 5 to 10 miles. The stereo can tell the landmark system that a specific landmark will be in a specific frame. (A frame is about 1 mile front to back and the bootstrap accuracy is better than 1/2 mile.)
- When the bootstrapper has lost a sequence of frames due to clouds. Position must be extrapolated using the wind and the airspeed, and the landmark subsystem must look in many frames for the landmark.

5.3.1 The Symchain Approach to Landmarks

The role and interactions of the Landmark Subsystem have been identified using an intensity-based landmark system developed under the Lockheed Independent Research Program. This approach identifies edges of objects and finds sets of connected edge points, called "symchains." The symchains are represented symbolically in terms of line segments and the angles between them, and this description is then compared with reference symchain descriptions in the landmark database. This edge-based approach, which was developed to deal with large perspective distortions, has the following characteristics:

The approach:

- o Is less sensitive to vehicle position/attitude errors than correlation-type approaches
- o Reduces the amount of storage necessary to match against a reference
- o Has the ability to obtain a match even when the reference is artificially obtained (as from a topographic map)

Characteristics which separate the approach from some other structural approaches are:

- o Extended "strands" are used in the match strategy, rather than linear edges being matched. These strands thus embody shape information which can be very useful in match environments.
- o Perspective effects can be compensated for, as the global match is performed in a later stage than the local matching. A camera transformation can thus be applied as a result of the matches in the different areas of the images.
- o The edge elements combined to form the strands are created using the grey level information in the image and must also be symmetrically associated with one another. (Each edge element in the strand considers its right- and left-hand edge associate as the best ones.) This provides robust determination of high gradient areas initially which simplifies the identification of edge structures.

The current approach to landmark navigation suffers from several shortcomings:

- o No consideration is given to the inherent shape constraints of well-chosen landmarks; e.g., the parallelism of river-banks and road-edges, the perpendicularity of crossroads, the anomalous height of water towers, transmission lines, etc.
- o Since the modeled features are essentially linear, it is only in the global matching of several features that one accounts for the inherent spatial relationships.
- o The global matching procedure which attempts to combine matched features does not use the available camera model information.

5.3.2 Operational Aspects of the Landmark Database

Before a vehicle using passive navigation can make an operational flight, it is necessary to load the landmark database with a symbolic description of landmarks that the vehicle will fly over. For this we require photographic imagery taken at some previous time from which landmarks on a 10-mile swath (representing the flight path) are chosen. It would be desirable to find suitable landmarks for every 5 miles along the swath so that the system could deal with the situation of obscured ground at any point in the flight path. A ground-based landmark analyzer processes the photographic data and converts the processed landmarks to symbolic form for loading into the vehicle memory. For each landmark, the processor indicates the best processing technique for the vehicle to use.

The Landmark Subsystem as used in the Passive Navigation system cannot be like a "barnstorming" pilot who looks out the window to check on landmarks. Because of the low altitude (say 1500 ft), the narrow field-of-view, and the look angles of 0 to 30 degrees from the nadir, the system is more like a pilot looking down through a hole in the floor of the aircraft. Note that looking at a swath 600 ft to 2400 ft in width has an important operational implication: We cannot set up a flight path and then look for landmarks adjacent to the flight path. Rather, we must set up the flight path so as to navigate from landmark to landmark, as shown in Fig. 5-1.

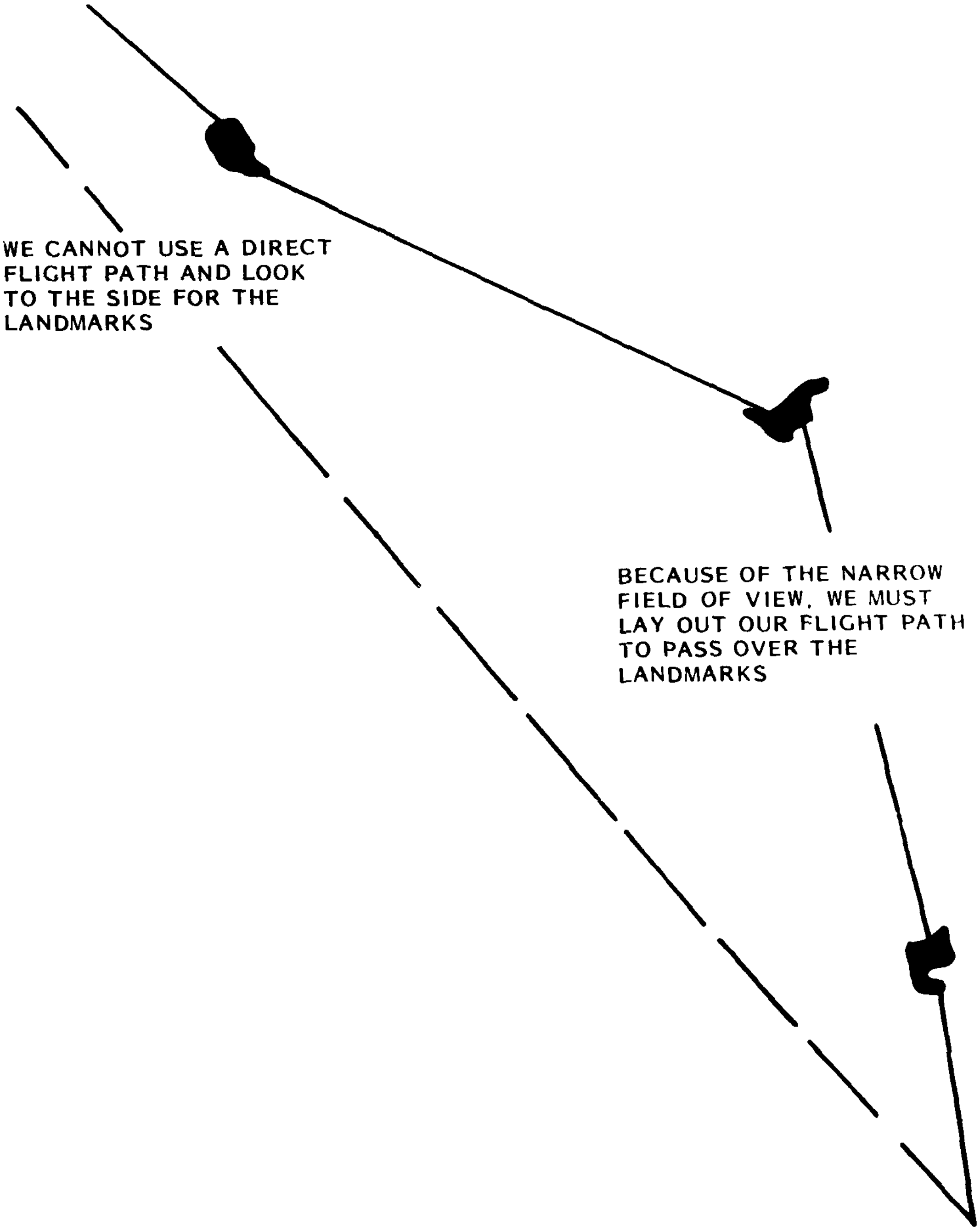

WE CANNOT USE A DIRECT FLIGHT PATH AND LOOK TO THE SIDE FOR THE LANDMARKS
BECAUSE OF THE NARROW FIELD OF VIEW, WE MUST LAY OUT OUR FLIGHT PATH TO PASS OVER THE LANDMARKS

Fig. 5-1 Planning a Flight Path in Passive Navigation

5.4 GENERAL DISCUSSION

Much work remains to be done before a passive navigation system based on images can become practical. Problems of speedup and mechanization, while difficult, can be accomplished. The problems remaining in stereo concern the accomodation of scale change and rotation, and the improvement in match point computation. The Landmark Subsystem requires further research in the representation of spatially distributed features such as parallelism (river, road), perpendicularity (cross roads), etc. Furthermore, the camera model should play a key role in guiding the global match.

A discussion of some of these future effects is given in Section 6.

Section 6
FUTURE WORK

The future work consists of both integrative (e.g., interfacing to a terrain model), and innovative tasks (e.g., extension to topographic landmarks). The general goals of future efforts would be to:

- Refine and extend the existing image-based navigation system
- Demonstrate the improved system using a suitable terrain model
- If actual flight data can be found, carry out navigation experiments using real data

Three main categories of future effort can be identified: (1) system aspects, (2) stereo subsystem, and (3) landmark subsystem.

6.1 SYSTEM ASPECTS

A "flight" over a suitable terrain model should be made using a calibrated vidicon camera* and known ground truth. These calibrated data would enable error analysis runs to be made.

It would also be useful to obtain real flight data so as not to develop algorithms which take advantage of the model characteristics but which are ineffective in the "real world." What is required is a sequence of 50 to 100 aerial photographs, having 65 to 75% overlap, and with a resolution of 5 to 10 feet per pixel. It is

*Calibration of a videon camera can be performed using a chart having an array of spots of known size and spacing. If the vidicon is a known distance from the chart, then the focal length and the distortion correction over the field of view can be computed as described in Ref. 19.

important that ground truth be available to relate items in the imagery to real points on the ground. If suitable imagery can be located, accuracy experiments using bootstrap stereo should be made.

As the various subsystems become better defined, it is possible to simulate the functions of the Executive in providing subsystem interaction: (1) in initiating recovery fallback procedures used when the navigation system becomes confused due to missing image sequences or poor advice from the subsystems, and (2) in combining outputs from the subsystems. Thus, future research should incorporate the type of expert judgment used by a navigator when he gets lost, e.g. "follow a major highway or river until a known landmark is found." The incorporation of such judgment is in the spirit of current "expert systems" being developed by the artificial intelligence community.

6.2 STEREO SUBSYSTEM

To obtain a more robust and efficient stereo subsystem, several important technical improvements must be made to the components of bootstrap stereo. These include:

o Accommodating Scale Change and Rotation. As the vehicle proceeds along its course, it will probably need to execute a variety of maneuvers for operational reasons. Changes in altitude or in the orientation of the vehicle will result in image-to-image scale changes and rotations. For moderate maneuvers, these changes will be small and will cause no problem with the correlation matching, but the bootstrapping procedure should also be able to handle more sudden maneuvers. Because the bootstrapper will have access to the vehicle's instruments, it will know the magnitude of any turns or elevation changes. From this information, it can calculate the necessary image rotations and scale factor changes to permit registration of the imagery.

- Improvement in the Match Point Calculation. We have carried out error simulation studies which show that it is important to attain the match point coordinates of corresponding stereo image-plane points to within 1/10 pixel or better, in order to maximize the number of bootstrap iterations obtainable before a landmark fix is needed. This requires interpolation of the correlation values, which can be calculated only at integer pixel displacements. At present, this is done by separate one-dimensional polynomial interpolation in each direction about the peak - a quick, but not highly accurate method. It would be important to incorporate more sophisticated correlation interpolation methods, such as least-squares fitting of a two-dimensional polynomial, into the bootstrapping code, and to perform experiments to determine if the desired level of match accuracy is being obtained.

- Computational Speedup Techniques. The present implementation of the bootstrapping code requires 5 minutes for analysis of one bootstrapping iteration. Speedup of the algorithm is necessary if demonstrations of the technique are to be run in reasonable amounts of time. (Deployment of the algorithm would, of course, require its implementation in hardware, which would result in considerable speedup over any software implementation.) A combination of programming improvements in the existing code and algorithm development is needed to obtain the desired speedup in the bootstrapping algorithms.

6.3 LANDMARK SUBSYSTEM

The landmark approach taken in our study to date has assumed independent landmark subsystem (LS) operation. An improved LS can be obtained by recognizing the inherent relationship between the bootstrap operation and that of the landmark subsystem. In particular, when a set of known ground points is related to a set of sensed points, the stereo camera model can determine the position of the vehicle. Thus, landmark-finding can be viewed as detecting a known set of points in a reference image so as to establish a correspondence between points on the reference and points in the sensed image. The bootstrap module then uses this information to obtain the positional update of the vehicle.

This important concept both simplifies the design and motivates further development of effective landmark primitives and local matching algorithms.

The integrated system would operate as follows:

- o LS extracts primitives from the sensed image
- o LS proposes a set of correspondences between the sensed and reference image points
- o LS transmits the real world and the image plane coordinates of the paired points to the camera model
- o The camera model in the Stereo Subsystem computes the vehicle position and transmits it to the Executive.
- o The Executive evaluates the results:
  - – If acceptable, the position estimate is used to correct the bootstrap estimate
  - – If the position estimates are unacceptable, the landmark subsystem is requested to propose a different set of corresponding points

To achieve an integrated LS, the following tasks must be addressed:

- o Expand the class of primitives to include river bends, road intersections, bridges, etc., by looking for corners and parallel lines. Investigate the use of depth map features as potential landmarks.
- o Speed up the current line extraction approach to take advantage of adjacency of edge points. (Since most of the edge points that become members of symchains are indeed adjacent, it is not necessary to complete the detailed operations of the general case.)
- o Develop a more robust description matching procedure.
- o Make use of the camera model to
  - – Accept or reject proposed matches between the sensed and reference feature pairs
  - – Compute the positional update for identified sensed features

Section 7
REFERENCES

1. M. J. Hannah, "Bootstrap Stereo," Proceedings: Image Understanding Workshop, April, 1980, College Park, MD, 30 Apr 1980

2. C. M, Bjorklund and D. L. Milgram, "Edge Structures for Image Matching," 1980 IEEE Workshop on Data Description and Management, Aug 1980

3. K. Stevens, K. Nishihara, B. Schunck, and the staff, "Understanding Images at MIT: Representative Progress," Proc. Image Understanding Workshop, U. of Maryland, Apr 1980

4. J. O. Limb, and J. A. Murphy, "Estimating the Velocity of Moving Images in Television Signals," Computer Graphics and Image Processing, 1975, pp. 311 - 327

5. R. O. Duda, and P. E. Hart, Pattern Classification and Scene Analysis, John Wiley and Sons, New York, New York, 1973

6. M. M. Thompson, Manual of Photogrammetry, American Society of Photogrammetry, Falls Church, VA, 1944

7. D. G. Gennery, "A Stereo Vision System for an Autonomous Vehicle," Proceedings of the 5th IJCAI, Cambridge, MA, 1977

8. M. A. Fischler, and R. C. Bolles, "Random Sampling Consensus," Proceedings: Image Understanding Workshop, College Park, MA, 30 Apr 1980

9. H. P. Moravec, "Visual Mapping by a Robot Rover," Proceedings of the 6th IJCAI, Tokyo, Japan, 1979

10. M. J. Hannah, Computer Matching of Areas in Stereo Imagery, Ph.D. Thesis, AIM #239, Computer Science Department, Stanford University, 1974

11. L. H. Quam, Computer Comparison of Pictures, Ph.D. Thesis, AIM#144, Computer Science Department, Stanford University, CA, 1971

12. R. Nevatia and K. R. Babu, "Linear Feature Extraction and Description," 6th ICJAI, Tokyo, 1979, pp. 639 - 641

13. S. W. Zucker and J. L. Mohammed, "A Hierarchical Relaxation System for Line Labeling and Grouping," Proc. IEEE PRIP Conference, Jun 1978, pp. 410 - 415

14. G. J. Vanderbrug and A. Rosenfeld, "Linear Feature Mapping," IEEE Trans. SMC, Vol. 8, Oct 1978, pp. 768 - 774

15. D. L. Milgram, "Edge Structures for Image Matching," 1980 IEEE Workshop on Data Description and Management, Aug 1980

16. C. M. Bjorklund and D. L. Milgram, "Graph-Theoretic Methods in Edge-Point Linking," 13th Asilomar Conference on Circuits, Systems, and Computer, Aug 1979

17. T. Pavlidis, Structural Pattern Recognition, Springer, Verlag, 1977.

18. K. R. Babu and R. Nevatia, "Use of Linear Features for Road Detection," Semi-Annual Technical Report, 31 Mar 1979, IPI, U. of Southern California, Los Angeles

19. H. P. Moravec, "Obstacle Avoidance and Navigation in the Real World by a Seeing Robot Rover," Stanford Artificial Intelligence Laboratory, Memo AIM-340, (Chapter 4, "Calibration") Sept 1980

Appendix A
NONCORRELATION VELOCITY DETERMINATION

An investigation of methods for obtaining the vehicle ground velocity while dealing with vehicle attitude changes led us to consider an approach that derived velocity from an analysis of the intensity changes from frame to frame. Using Lockheed Independent Research funds, a concept using a linear array was developed, and was reported on in an Image Understanding Workshop and in an SPIE paper. Because of the relevance of this concept to Passive Navigation, the paper is included as Appendix A.

219-27

Airborne Ground Velocity Determination by Digital Processing of Electro-Optical Line Sensor Signals

Moshé Oron*

Department of Aeronautical Engineering
Technion, Israel Institute of Technology
Haifa, Israel

and

O. Firschein
Lockheed Palo Alto Research Laboratory
Department 52-53, Building 204
3251 Hanover Street
Palo Alto, CA 94304

Abstract

Signals from a solid state electro-optical line sensor, which samples a two-dimensional image brightness function in time and space, can be digitally processed to extract the ground velocity vector of relatively slow, autopilot-controlled aircraft such as mini-RPVs. This sensor can be rotated into the direction of motion by a stepping motor which is controlled by a computational unit using simple easily realizable algorithms to keep the sensor in alignment with the velocity vector as well as to compute its magnitude. Together with other instruments already installed onboard the aircraft, this combination of sensor and computational unit may form an instrumentation setup which can be used in passive, autonomous navigation systems. Computer simulated experimental runs proved that a sufficient degree of directional sensitivity and overall accuracy can be attained with the proposed method.

Introduction

Continuous on-board determination of ground velocity followed by time integration, can be used for autonomouse vehicle navigation. While realizing such a dead-reckoning system, it is necessary to minimize error accumulation, mainly due to changes in the vehicle's attitude. If a two-dimensional imaging device such as a TV camera were used for the velocity determination, it would be necessary to mount it on a gimballed inertial platform similar to those used in accelerometer-based navigation systems. This could result in an instrumentation package which might be more massive, expensive and limited in application than conventional Inertial Navigation Systems (INS), but less accurate. Image-based velocity determination for navigational purposes can, therefore, become practical only if it is realized at much lower cost and weight than INS. This is particularly true when considering the most likely types of aircraft in which such systems would be mounted in order to provide them with autonomous passive navigation[1] capabilities: the miniature Remotely Piloted Vehicles (mRPV), some of which are less expensive than an advanced INS.

In the system described here, simplicity, use of dimensionally small and inexpensive components, and exploitation of other instruments usually installed in an mRPV are emphasized. A solid state electro-optical line sensor, such as the linear array CCD which is inherently small and relatively inexpensive, is used for imaging. Since this sensor is amenable to electronic compensation of attitude changes of autopilot controlled aircraft, as has been recently demonstrated by Oron and Abraham[2], the necessity for electromechanical and gyroscopic stabilization of the optical axis is eliminated thus avoiding a heavy cost and weight penalty.

An additional advantage of one-dimensional imaging is in the much lower rate of data generation as compared with the two-dimensional case. This makes it possible to use modest computational power and limited storage memory in the digital signal processing phase. Furthermore, a new computational method which is based on brightness differences, rather than on two-dimensional correlation, is used to extract the velocity vector from a moving scene. The new method is not only less scene-dependent, and therefore less limited in scope, but also requires significantly less time and memory.

Basic concept

A vertical cross section through an airborne imaging system based on a line sensor containing M elements, each giving rise to a pixel (picture element) of size δ and intensity $I_{n,i}$, is shown in Figure 1. The i-index ($i = 1,\dots,M$) designates the position of the element (or pixel) along the sensor axis, x, which lies in the focal plane of the lens, at an angle γ to the longitudinal axis of the aircraft, x_a (see Figure 2). The n-index ($n = 1,\dots,N$) designates the pixel's time of occurrence:

$$t_n = n\tau \qquad (1)$$

* During 1979-80 Academic Year: Visiting Scholar at Department of Aeronautics and Astronautics, Stanford University, CA 94305

where τ is the exposure time of the sensor between readings. Thus, every τ milliseconds the sensor generates M pixels or readings of intensity $I_{n,i}$ which are discrete samplings of a two-dimensional image brightness function, f(x,y), taken along the x-axis at time t_n. Since x and y vary with time because of the aircraft's motion relative to ground (the ground velocity, V), the intensity readings will also vary with time.

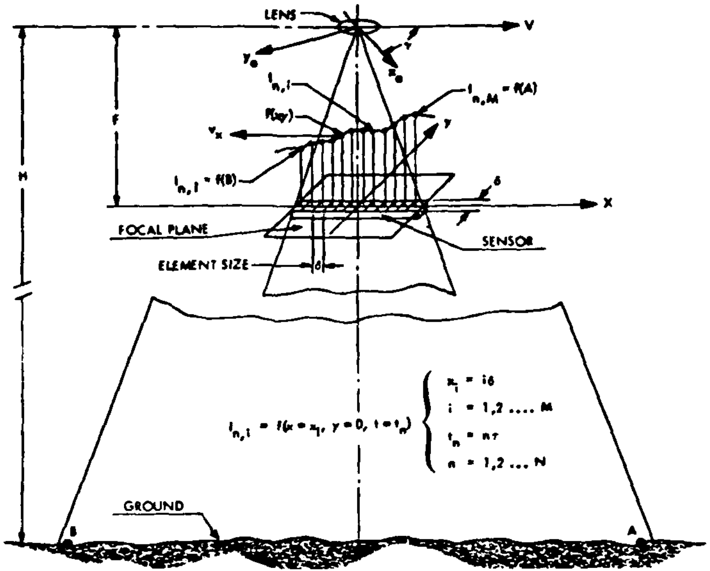


Figure 1. Vertical cross section of airborne imaging system (sensor is parallel to ground velocity vector: $\rho = 0$)

It is easier to visualize this variation as being caused by a motion of the whole brightness function f(x,y) in the focal plane in a direction opposite to that of the ground velocity, V. This "focal velocity" is given by:

$$v = -\frac{F}{H} \cdot V \tag{2}$$

where F is the focal length of the lens and H is the aircraft altitude relative to ground. Since F is constant and known and H is measured independently by other on-board instrumentation, the extraction of v from the $I_{n,i}$ readings will make it possible to continuously determine and integrate V with respect to time.

As mentioned above, f(x,y) is an implicit function of t. For those parts of this function which are also continuous and analytic, it is possible to calculate the total time derivative:

$$\frac{df}{dt} = \frac{\partial f}{\partial x} \cdot \frac{dx}{dt} + \frac{\partial f}{\partial y} \cdot \frac{dy}{dt} \tag{3}$$

From Figure 2 and equation (2) the following relations are derived:

$$\frac{dx}{dt} = v_x = -\frac{F}{H} V_x ; \frac{dy}{dt} = v_y = -\frac{F}{H} V_y \tag{4}$$

$$v = \sqrt{v_x^2 + v_y^2} \tag{5}$$

$$\tan \rho = \frac{v_y}{v_x} \tag{6}$$

Substituting (4) in (3), an expression for v_x is obtained:

$$v_x = \frac{\frac{df}{dt} - \frac{\partial f}{\partial y} \cdot v_y}{\frac{\partial f}{\partial x}} \tag{7}$$

For the simple case of $v_y = 0$, i.e. when the sensor is aligned exactly along the ground velocity vector ($\rho = 0$), equation (7) becomes:

$$v = v_x = \frac{\frac{df}{dt}}{\frac{\partial f}{\partial x}} \tag{8}$$

The time and space derivatives in (8) can be approximated by brightness difference expressions, designated $\Delta^n I_i$ and $\Delta^i I_n$ respectively. The $\Delta^n I_i$ expression is related to the total time derivative at $x = x_i$ and is calculated from successive intensity readings ($I_{n-1,i}$, $I_{n,i}$, $I_{n+1,i}$ ...) of the i^{th} element:

$$\frac{df}{dt} = \frac{\Delta^n I_i}{\tau} \simeq \frac{I_{n,i} - I_{n-1,i}}{\tau} \tag{9}$$

Similarly, $\Delta^i I_n$ is related to the partial space derivative, or brightness gradient, calculated at time $t = t_n$:

$$\frac{\partial f}{\partial x} = \frac{\Delta^i I_n}{\delta} \simeq \frac{I_{n,i} - I_{n,i-1}}{\delta} \tag{10}$$

Substitution of (9) and (10) into (8) while accounting for the optical sign reversal between V and v leads to:

$$v_{n,i} = -\frac{\Delta^n I_i}{\Delta^i I_n} \cdot \frac{\delta}{\tau} \tag{11}$$

where $v_{n,i}$ is the focal velocity calculated at the i^{th} element at time t_n. In order to simplify (11) and eliminate the minus sign, a negative unit vector velocity, v_p, equal to one pixel, δ, per one exposure time, τ, in the minus x direction is defined:

$$v_p \triangleq -\frac{\delta}{\tau} \tag{12}$$

and substituted into (11) yielding:

$$v_{n,i} = \frac{\Delta^n I_i}{\Delta^i I_n} \cdot v_p \tag{13}$$

A further simplification is obtained by defining a dimensionless focal length velocity coefficient, $\nu_{n,i}$ as follows:

$$\nu_{n,i} \triangleq \frac{v_{n,i}}{v_p} = \frac{\Delta^n I_i}{\Delta^i I_n} \tag{14}$$

Using equation (14), a total of $(M - 1)\cdot(N - 1)$ values of $\nu_{n,i}$ are obtained during one period, T, of velocity estimation, where T is given by:

$$T = N \cdot \tau \tag{15}$$

If M and N are large enough, a statistically satisfactory distribution of the $\nu_{n,i}$ values is achieved enabling a good estimate of the true value of ν. One simple such estimate would be the arithmetic mean:

$$\bar{\nu} = \frac{1}{(M - 1)(N - 1)} \sum_{i=2}^{M} \sum_{n=2}^{N} \frac{\Delta^n I_i}{\Delta^i I_n} \tag{16 }$$

The discussion of more elaborate statistical procedures is out of the scope of this paper and will be presented elsewhere[3].

Proposed instrumentation setup

For the implementation of the basic concept described above, an instrumentation setup which includes some components normally found in RPVs is proposed (Figure 3). Employing the CCD line sensor (1024 elements) used by Oron and Abraham[2] in their system ($\delta = 12.5$ µm, $\tau = 2$ msec) with optics similar to theirs ($F = 100$ mm) but arranged in an imaging geometry shown in Figure 1; a negative unit-pixel velocity $v_p = -6.25$mm/sec is obtained. The focal velocity for typical flight conditions ($V = 80$ kt, $H = 3000$ ft) is $v = -4$ mm/sec and the velocity

219-27

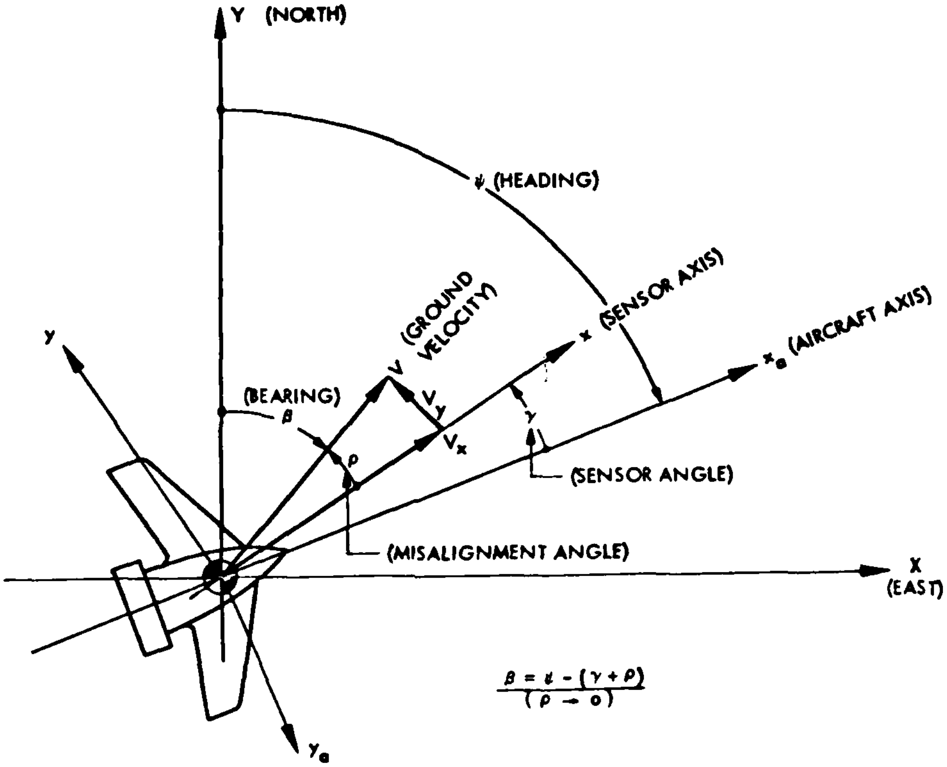


Figure 2. The various axis systems projected onto the ground plane.

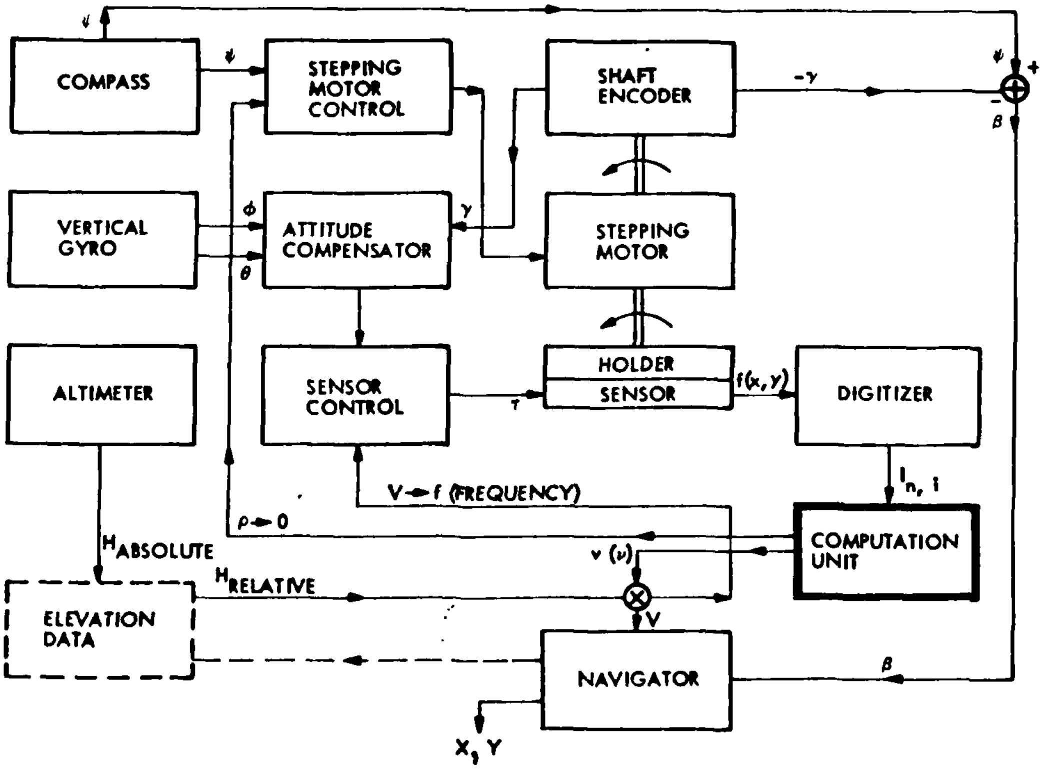


Figure 3. Block diagram of proposed instrumentation setup.

219-27

coefficient is $\nu = 0.64$. As the ground velocity will vary between 40 knots to 120 knots, the variation of this coefficient will be $\nu = 0.32 - 0.96$. In other words, the translation along the sensor between two adjacent exposures will be between about a third of a pixel and just less than a whole pixel. The translation per exposure due to pitch and roll instability is less than 10% of that, but its accumulation can be compensated for using signals from the on-board vertical gyro installed in the autopilot system of the RPV as previously described[2]. For that purpose only 512 readings out of the 1024 elements will be actually processed in the computational procedure for determining ν .

The 511 interpixel differences, $\Delta^i I_n$, will be computed on-line during one exposure time τ (on the order of 2 msec) and transferred to a special register together with the 512 values of I_i . At the end of the second exposure time ($n = 2$) the 511 interexposure differences $\Delta^n I_i$ will also be computed using equation (9). Division of the latter set of brightness differences by the former will yield a maximum of 511 values of ν after every exposure. In reality, the computation unit shown in the block diagram of the instrumentation set up (Figure 3), will also include a logical circuit which will discard many of the above computations: the $\Delta^n I_i$ denominator values which are lower than a certain threshold, will first be eliminated in order to prevent erroneously high ν results caused by scenery which is too low in contrast ("flat") to yield a meaningful spacial brightness gradient. Secondly, very high or abrupt gradients ("sharp edges") will also be discarded since they represent discontinuities in the $f(x,y)$ brightness function or in its first derivative and, therefore, belong to non-analytical portions of $f(x,y)$. If a period of about $T = 80$ msec ($N = 40$) is chosen, over 20,000 values of ν will be computed, and at least 10,000 of them are expected to be valid for each period. This is a relatively large and statistically controlled population of results which assures a Gaussian distribution, provided that only random errors (or noncorrelated noise) occur in the system while all the systematic errors are eliminated.

An important systematic or non-random error may arise as a result of misalignment between the sensor and the direction of the ground velocity vector as can be seen from comparison of equations (7) and (8). To prevent such misalignment, the sensor could be mounted on a special holder which will be rotated in the focal plane around an axis of rotation which is perpendicular to the earth surface or parallel to the z axis of the aircraft. The rotational motion can be provided by a small stepping motor which need not be faster than 100 Hz (10 msec per step) or more accurately positioned than within 1°. Thus, if the direction of motion is known, and the time spent at each angle is 80 + 10 = 90 msec, the system will during one second check about 10° of angle, 5° to each side of this direction. After such angular scanning, about 40 different sets of ν computations, each consisting of more than 10,000 valid values will be obtained. The set which will have the best statistical distribution of its values around the mean (the most Gaussian-like and symmetric histogram), will be the one with the smallest systematical or directional bias error and, therefore, the sensor will then be most closely aligned with the ground velocity direction.

Mounting of the electro-optical sensor on a rotational holder has an additional practical advantage: it enables compensation for sudden changes in the yaw or heading angle ψ measured by an on-board compass. The output of this instrument will be processed to provide a correctional signal which can be fed into the stepping motor controller. Thus, whenever a sudden change in yaw occurs, mainly due to gusts of wind, the stepping motor will compensate for it immediately (a 100 Hz rate is much faster than required by the flight dynamics of a mRPV).

So far only direction-following or tracking was discussed. The problem of initially setting up the sensor along the direction of the ground velocity vector can be solved within 5 to 10 seconds. In the worst possible case, a "brute force" scan at a rate of 20° of angle per second will require 9 seconds to check all possible directions (180° of angle) to an accuracy of 2°, but it is more likely that the ground velocity direction will be found sooner than that. In fact, a strategy of starting from the heading direction of the aircraft and scanning systematically on its both sides, will lead much faster to the plane's bearing angle β (which gives the ground velocity direction relative to the North) since in most cases the heading and bearing of a mRPV are not too widely separated from each other.

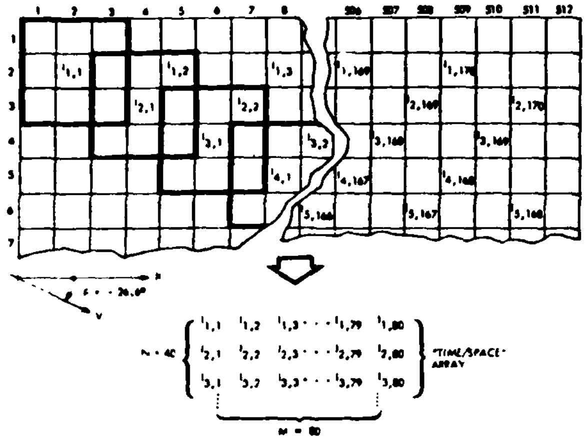
Computer-simulated experimental investigation

This section describes a computer simulation of vehicle motion, the computational procedures for carrying out the simulation experiments, and the results obtained.

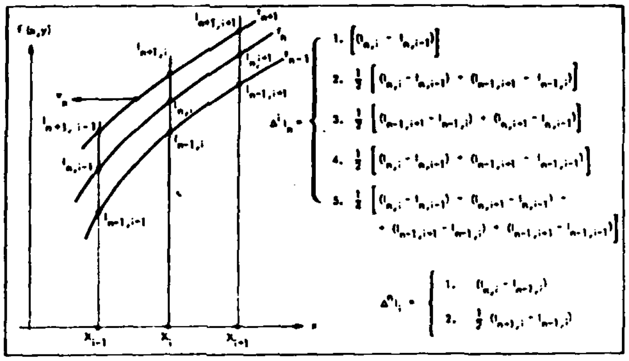
Simulation of vehicle motion

As discussed previously, the speed of the vehicle and the optics used result in focal velocity magnitudes which are less than one pixel per exposure ($0 < \nu < 1$). In order to simulate such sub-pixel motions with a computer, digitized aerial photographs 512 × 512 pixels in size, at about 0.5 m/pixel resolution, were used. By convolving the original photographs with a 3 × 3 equally weighted window, as shown in Figure 4, new digital images 170 × 170 pixels in size, at about 1.5m/pixel resolution, were formed. By selectively sampling the convolved images, motion in 1/3 pixel increments per exposure can be simulated (a $\nu_x = 2/3$ and $\nu_y = -1/3$ motion is demonstrated in Figure 4). These simulations finally generate the $I_{n,i}$ readings which are stored in the computer as N,M "time/space" arrays (n = row index, i = column index), usually 40 × 80 in size

219-27



1 2 3 4 5 6 7 8 506 507 508 509 510 511 512
1 2 3 4 5 6 7
$I_{1,1}$ $I_{1,2}$ $I_{1,3}$ $I_{1,169}$ $I_{1,170}$
$I_{2,1}$ $I_{2,2}$ $I_{2,169}$ $I_{2,170}$
$I_{3,1}$ $I_{3,2}$ $I_{3,168}$ $I_{3,169}$
$I_{4,1}$ $I_{4,167}$ $I_{4,168}$
$I_{5,166}$ $I_{5,167}$ $I_{5,168}$
$\rho = -26.6°$
x
v
N = 40
"TIME/SPACE" ARRAY
M = 80



f(x,y)
$I_{n+1,i+1}$
I_{n+1}
$I_{n+1,i}$
I_n
$I_{n,i+1}$
I_{n-1}
$I_{n-1,i+1}$
v_x
$I_{n+1,i-1}$
$I_{n,i}$
$I_{n,i-1}$
$I_{n-1,i}$
$I_{n-1,i-1}$
$\Delta^i I_n =$
1. $[(I_{n,i} - I_{n,i-1})]$
2. $\frac{1}{2}[(I_{n,i} - I_{n,i-1}) + (I_{n-1,i+1} - I_{n-1,i})]$
3. $\frac{1}{2}[(I_{n-1,i+1} - I_{n-1,i}) + (I_{n,i+1} - I_{n,i-1})]$
4. $\frac{1}{2}[(I_{n,i} - I_{n,i-1}) + (I_{n-1,i+1} - I_{n-1,i-1})]$
5. $\frac{1}{4}[(I_{n,i} - I_{n,i-1}) + (I_{n,i+1} - I_{n,i-1}) + (I_{n-1,i+1} - I_{n-1,i}) + (I_{n-1,i+1} - I_{n-1,i-1})]$
$\Delta^n I_i =$
1. $(I_{n,i} - I_{n-1,i})$
2. $\frac{1}{2}(I_{n+1,i} - I_{n-1,i})$
x_{i-1} x_i x_{i+1} x

Figure 4. Formation of the "time/space" array by 3 × 3 window convolution and subsequent motion of $v_x = 2/3$, $v_y = -1/3$ ($|v| = 0.745$, $\rho = -26.6°$)

Figure 5. Five difference expressions, $\Delta^i I_n$, for space derivative and two for the time derivative, $\Delta^n I_i$.

($N = 40$, $M = 80$). For finer variation in v_x and v_y a larger convolution window can be used (e.g. for 5 × 5 the v increment is 0.2).

Computational procedures

After generating the time/space arrays and before performing any computations, the $I_{n,i}$ values were normalized and stretched with the minimum reading set equal to zero and the maximum to 100. After a period of preliminary experimentation, it became apparent that in order to approach the required condition of f(x,y) being continuous and analytic, a one-dimensional smoothing operation had to be performed on the rows of the $I_{n,i}$ array. Best results were obtained using a convolution-type smoothing with weighted one-dimensional windows, 5 to 21 elements long. The weight coefficients depend on the length of the window, as described by Sawitzky and Golay[4].

The most important computational procedure that had to be developed concerned the calculation of the time and space derivaties of f(x,y) using the $I_{n,i}$ readings. The two simplest $\Delta^i I_n$ and $\Delta^n I_i$ difference expressions were presented in equations (9) and (10). Four more for the space difference, $\Delta^i I_n$, and one more for the time difference, $\Delta^n I_i$, (see Figure 5) were experimented with using known analytical f(x) functions approximated by discrete samplings and obscured with random noise. Best results were obtained by using the combination of:

$$\Delta^i I_n = \frac{1}{2}[(I_{n,i} - I_{n,i-1}) + (I_{n-1,i+1} - I_{n-1,i})] \qquad (17)$$

for the space difference expression, and equation (9) for the time difference expression. This combination was therefore used in the experimental runs which followed.

Experimental runs and results

Most of the experimental runs were performed on the aerial photograph shown in Figure 6. This rural scene was chosen deliberately as one which represents a worse than average case since it is not only quite monotonous and "flat" containing little detail in the form of contrast and shape variations, but also displays a repetitive pattern of trees which makes it a difficult case for two-dimensional correlation methods. Figure 7 summarizes nine systematic runs which were performed after convolution with a 5 × 5 window for nine different ρ values changing from 90° through 0° to -90° with $|v|$ varying between 4/5 = 0.8 and 0.89. The nine computed v histograms (each of a population of about 2000 values) show a predicted behavior: the one for $v = v_x = 4/5$ ($\rho = 0°$) is almost perfectly Gaussian, displaying a sharp symmetrical peak, while the others are skewed with a pronounced decrease in peak height, sharpness and symmetry as $|\rho|$ increases. To further investigate the dependence of the distribution of v on the angle ρ, convolution windows of 5 × 5, 7 × 7,

219-27

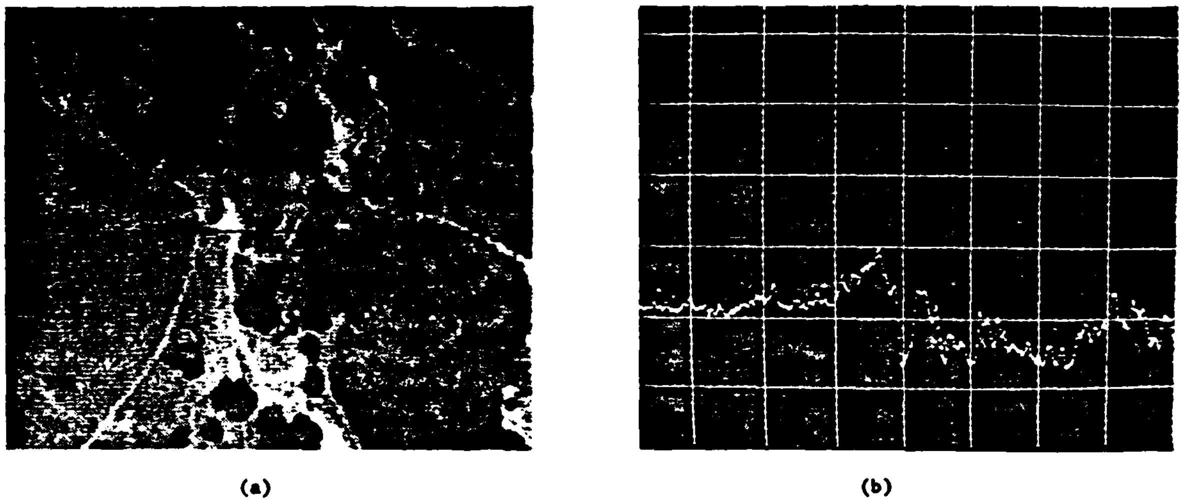


Figure 6. Example of 512 × 512 digital image of aerial photograph used in the computer experiments (a), with intensity curve (b), of horizontal line indicated by black cursor in (a).

9 × 9, 11 × 11, 15 × 15 and 21 × 21 were used, achieving minimum misalignment angles of about 11°, 8°, 6°, 5°, 3.5° and 2.5°, respectively. Since it was not practical to further increase the convolution window without introducing artifacts as well as drastically reducing the population size of the computed υ values, the behavior of the distribution for smaller angles had to be extrapolated as shown by the dotted line of Figure 8, which is a plot of the variation of the relative modal statistical frequency with the misalignment angle ρ .

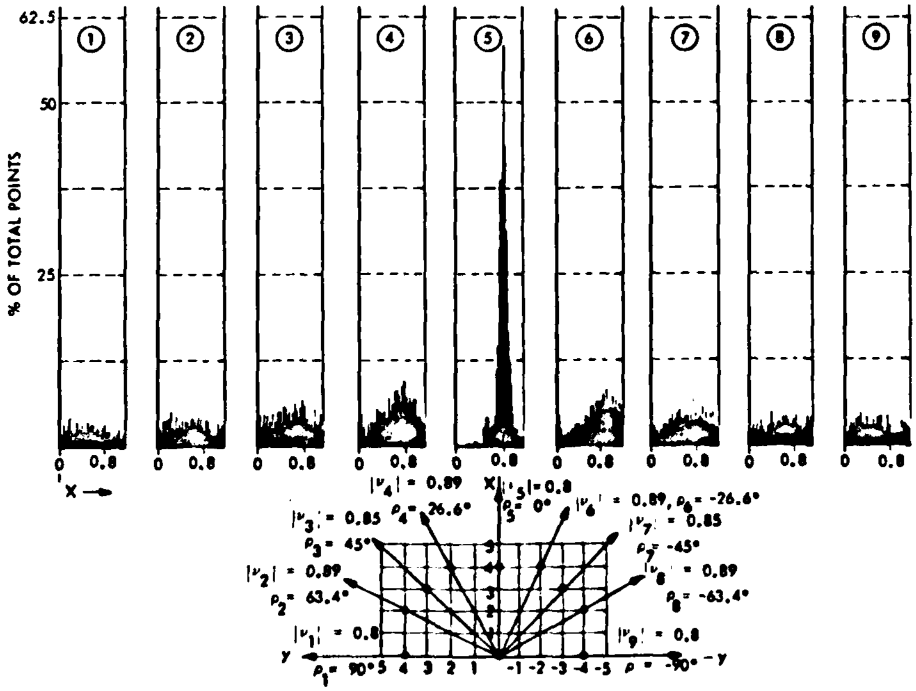


Figure 7. Computed velocity histograms for nine misalignment angles varying from $\rho = 90^{\circ}$ through $\rho = 0^{\circ}$ to $\rho = -90^{\circ}$.

Discussion

From Figures 8 and 9 it is evident that while working at ν values close to unity, it is possible to achieve high directional sensitivity as well as magnitude accuracy. In order to approach such ν values it is necessary to introduce a change in τ as V varies. This is shown in Figure 2: after V has been calculated, a signal is fed-back to the sensor controller which changes its frequency between 200 KHz and 500 Hz according to Table 1.

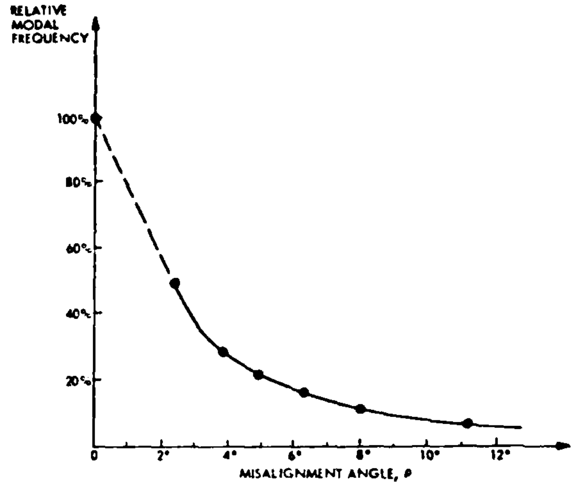


Figure 8. Plot of relative modal frequency of histograms versus misalignment angle ρ .

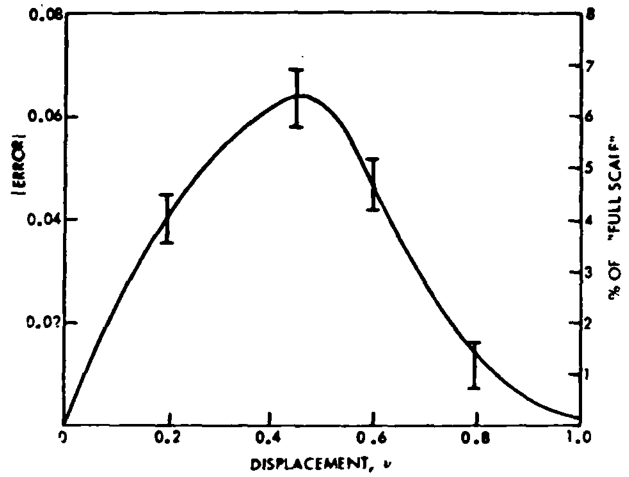


Figure 9 Plot of absolute error and % F.S. (F.S. = 100% for ν = 1.0) of ν versus the magnitude of ν .

Table 1. Sensor controller frequency (f) for varying aircraft speed (V)

V(knots)	f(KHz)	τ(msec)	v_p(mm/sec)	ν
40 - 49	200	5.0	2.5	0.8 - 0.98
50 - 62	250	4.0	3.125	0.8 - 0.99
63 - 75	300	3.3	3.75	0.83 - 0.99
76 - 85	350	2.9	4.375	0.88 - 0.99
86 - 99	400	2.5	5.0	0.86 - 0.99
100 - 110	450	2.2	5.625	0.89 - 0.98
111 - 120	500	2.0	6.25	0.91 - 0.98

From further examination of Figure 3 it becomes evident that the overall accuracy of the ground velocity determination, and hence of the whole dead-reckoning navigational system, is perhaps more critically dependent on the auxiliary readings of the altitude, H, and the heading angle, ψ, than on the new electro-optical sensor based part of the system. If a passive absolute altitude barometric altimeter is used, an accuracy of approximagely 1% can be achieved, however the uncertainty in terrain elevation data prestored in the system and subtracted from H to yield relative altitude,may raise that figure to a 2% level which for the $\nu \simeq 0.9$ range (see Figure 9) is worse than the ν accuracy (about 1%). From Figure 8 it is possible to deduce that an accuracy of about 1^{o} can be achieved for the angle of velocity, compatible with the compass ψ readings. Further experiments with hardware rather than computer simulated motion are needed to establish whether a better accuracy in γ can be accomplished justifying the use of more accurate instrumentation for ψ measurements.

In summary, it can be stated that ground velocity determination on-board mRPVs using electro-optical line sensors in combination with existing instruments is not only feasible, but quite practical, sufficiently accurate and not too expensive (in terms of weight and cost). The velocity values are computed at a rate of between 10 to 5 per second which is much higher than necessary for the relatively slow flight dynamics of a mRPV. This enables, by using fairly simple filtering and prediction techniques in the navigator prior to integration, to raise the level of confidence in this dead-reckoning navigation system as well as to overcome short "dark" periods when, due to cloud coverage, or extremely "flat" scenery, no velocity computations can be made. In this context it should be emphasized that the method proposed here is by no means limited to the visual spectrum; any electro-optical line sensor can be used, thus perhaps expanding the range of applicability to include overcast days as well as night operation.

Acknowledgments

The experimental work was performed in the Signal Processing Laboratory of the Lockheed Palo Alto Research Laboratory. We would like to thank Dr. J. J. Pearson, head of the group, for his favorable attitude which made it possible to complete this study, and other members of the group, especially Dr. W. G. Eppler, Dr. K. Dutta, and Mr. C. D. Kuglin for many helpful discussions and suggestions. One of us (M.O.) would like to thank Professor D. DeBra, head of the Flight Control and Instrumentation Laboratory of the Department of Aeronautics and Astronautics, Stanford University for his kind hospitality and most constructive comments on this presentation. The authors would like to express their gratitude to L. Staley for speed and accuracy in the preparation of the several drafts and the final manuscript.

References

1. Firschein, O., Gennery, D., Milgram, D., and Pearson, J. J., "Progress in Navigation Using Passively Sensed Images," Image Understanding Workshop, Menlo Park, CA, April 1979.

2. Oron, M., and Abraham, M., "Analysis Design and Simulation of Line Scan Aerial Surveillance Systems," Proc. SPIE, Vol. 219, 1980.

3. Oron, M. and Firschein, O., "Digital Processing of Images for Extraction of Translational Motion," (in preparation for the J. of Image Processing and Computer Graphics).

4. Savitzky, A. and Golay, M., "Smoothing and Differentiating of Data by Simplified Least Squares Procedures," Analytical Chemistry, Vol. 36, No. 8, July 1964, p. 1627-1677.

Appendix B
IMAGE VELOCITY SENSOR RESULTS

B.1 IMAGE VELOCITY SENSOR

The original concept for the Passive Navigation System utilized the Lockheed Image Velocity Sensor (IVS), a device that uses phase correlation to obtain the offset between two successive frames (Ref. B-1). From the offset and the time between frames, it is possible to obtain the velocity of the vehicle, and thus perform dead reckoning navigation. A series of experiments using the NVL terrain model imagery was carried out using a pair of 128 x 128 pixel images that were then processed on the Lockheed IVS processor.

It was found that the camera was forward pointing and not aimed along the flight path. This resulted in large perspective effects in the images; from frame to frame, objects at the top of an image moved less than those at the bottom. In addition, objects on the left side of the image moved parallel to the direction of motion, while those on the right side translated from left to right. As shown in Fig. B-1, the IVS displacements depended on the portion of the image from which the 128 x 128 subregion was selected. This effect occurred even for images having 80% overlap.

Thus, we found that if the vehicle pitch and roll are large enough to cause perspective effects, the IVS should use a 128 x 128 image covering a large field-of-view to compensate somewhat for the offset errors. This was verified by compressing the 512 x 512 image (by sampling to 128 x 128) and running the experiment again.

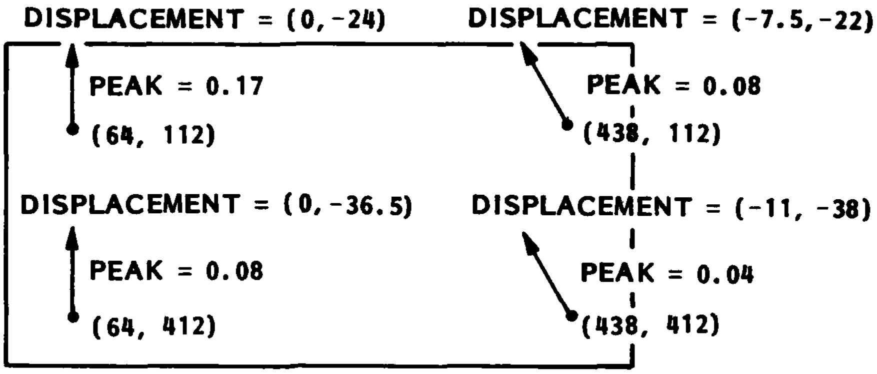


Fig. B-1 Image Velocity Sensor Offsets Obtained from Various Portions of the Image

Because of the vulnerability of the IVS approach to vehicle pitch, roll, and yaw, it was decided to use a stereo-based approach with camera transformations that deal with vehicle perturbations. Prior to the bootstrap stereo concept, we considered a camera mounted in each wing to obtain the stereo pair. Unfortunately, due to the short baseline between cameras, wing flexure became an important problem.

B.2 STRUCTURAL FLEXURE COMPENSATION

A study was made using stereo derived from two cameras, mounted on opposite wings or mounted fore and aft on the fuselage. If we are dealing with flight altitudes on the order of 1000 ft and cameras that are at least 10 ft apart, a 1% accuracy in altitude determination requires a resolution of 0.1 mrads. When dealing with this accuracy in resolution, the relative camera orientation change due to vehicle structural flexure becomes important. Various techniques for measuring the camera-relative orientations were investigated, including the sensor cluster approach shown in Fig. B-2.

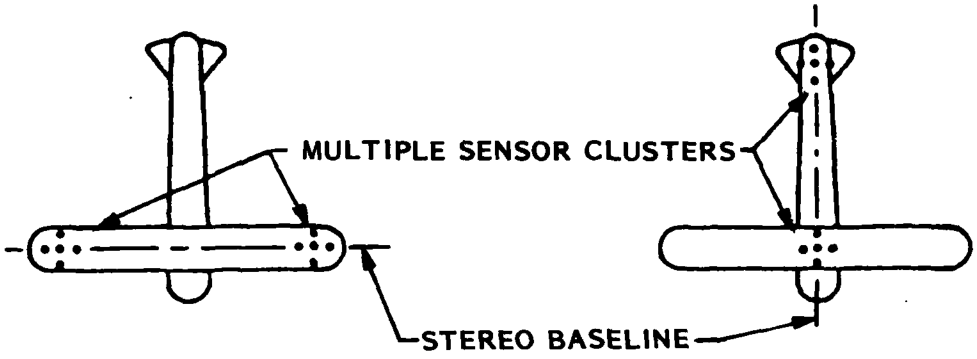


Fig. B-2 Location of Sensor Cluster on Vehicle

In the cluster calibration approach, a cluster of sensors having a narrow field-of-view is used at each position, as shown in Fig. B-1. A series of experiments was made for the various cluster forms shown in Fig. B-1, and the following table shows the resulting pan accuracy due to the uncertainty in the point group positions, and the pan error resulting from a scale factor error of 1 part in 1000 in the point spacings. (With single cameras, this would correspond to an error in one focal length of 1 part in 1000.) The values are in degrees.

Point Group	Pan Error From Points	Pan Error From Scale
3A	0.0109	0.058
3B	0.0109	0.058
4A	0.0091	0.055
4B	0.0029	0.00057
5	0.0029	0.00057

The roll error in all of the above cases was less than 0.02 deg. In Ref. B-2 we determined that a maximum pan error on the order of 0.0057 deg. and a maximum roll error on the order of 0.29 deg. was allowable. We can therefore see that the last two cases are satisfactory, whereas the first three are unsatisfactory with the assumed scale factor accuracy and would still be questionable even if the scale were known exactly.

Therefore, it seems safe to assume that at least four points will be needed with this method. Furthermore, the point or points for the main height measurement should be directly under the cameras. Since Group 4B does not contain such a point, an extra pair of cameras would seem to be required with it, and its results might as well be used in the camera model determination. Therefore, Group 5 apparently is the best arrangement to use, since it contains such a point. The total effect on height accuracy of the camera model determined from Group 5 is 5.0 ft. Combined with the point accuracy of 10 ft, this produces a total accuracy of 11.2 ft.

With the development of the bootstrap stereo concept that uses a single camera, the multicamera approach described above was abandoned.

B.3 REFERENCES

B.1 O. Firschein and J. J. Pearson, "Artificial Intelligence Concepts Applied to Navigation Using Passively Sensed Images," Proc.: Image Understanding Workshop, Pittsburgh, Penna., 14-15 Nov 1978

B.2 Passive Navigation R&D Status Report No. 2, 15 Sep 1978, Lockheed Palo Alto Research Laboratories

APPENDIX C

STEREO VISION AND LINEAR FEATURE TRACKING
FOR PASSIVE NAVIGATION

October 1980

T.O.Binford and the staff

Image Understanding group
Artificial Intelligence Laboratory
Computer Science Department
Stanford University
Stanford, California, 94305

[1]

Introduction

The objective of this research is to develop advanced techniques which contribute to passive navigation by imaging sensors. This report describes progress on two capabilities which are central to robust passive navigation: first, tracking extended linear features, and second, accurate registration of sequences of images of buildings and cultural structures.

Tracking extended linear features promises to decrease computation and storage requirements while increasing reliability. Linear feature tracking involves first building image descriptions with curvilinear elements then matching image descriptons of linear features.

Research at Stanford and elsewhere has resulted in systems which demonstrate reasonable capabilities for stereo mapping of smooth terrain and for terrain following. Registration of image sequences of buildings and cultural structures is a particular problem for area correlation methods of stereo mapping and image matching.

Buildings and structures are difficult because their surfaces are discontinuous and have no texture, and because buildings have vertical surfaces. Also, buildings are regular, i.e. aligned in direction and in grid patterns. Relief introduces disparities which affect image registration. No effective and efficient way within area correlation has been found to segment surfaces accurately at discontinuities. Systems must provide ways to map textureless areas where there is no information in the image to match by correlation and areas where there are ambiguous local matches.

Our research has concentrated on: 1. stereo mapping using edge features to provide segmentation of surfaces at discontinuities; 2. surface interpretation to provide a means of interpolation where no depth information is available; 3. geometric constraints on surface interpretation to aid in choosing globally consistent solutions among locally ambiguous solutions.

Research on passive navigation was made in collaboration with other research on stereo vision and image description with edges supported by the ARPA Image Understanding program and the National Science Foundation. The combined effort involved Dr. Tom Binford, Prof. P.J.MacVicar-Whelan, Michael Lowry, and Peter Blicher on edge description, and Binford, Dr. Sidney Liebes, Jr., R. D. Arnold, and Harlyn Baker on stereo vision. The Passive Navigation subcontract directly supported about 8 months of Dr. Liebes effort and 8 months of Michael Lowry's time. Only salaries were included in the subcontract. Computing facilities, project direction, communications and other expenses were covered by the main Image Understanding contract. Substantial computing facilities were made available to Lockheed staff and substantial aid in evaluating Gennery's stereo mapping system was provided by Donald Gennery.

Summary

We have made substantial progress in improving edge operators based on principles introduced in Binford-Horn and Herskovits and Binford. We have found novel and improved methods of localization of edges. We have implemented a simplified operator based on Binford-Horn and show results from it. Although we have not yet completed a system which could perform the tasks required for passive navigation by linear landmark features, we plan to integrate these results in a system which will do those tasks.

We have made theoretical analysis of geometric constraints for special surfaces which occur in buildings and cultural structures. We have analyzed geometric constraints of horizontal and vertical planes and cylinders, especially vertices of orthogonal triples. These constraints are especially important to augment general stereo vision capabilities.

We have made a new edge-based stereo vision system which was tested on images from the NVL sequence. The stereo system appears to give relatively accurate terrain maps and could be used in a stereo TERCOM approach.

We have analyzed geometric constraints for general stereo vision and have introduced several strong new constraints which will have a significant impact on performance of stereo vision systems. We are in the process of extending the system of geometric constraints and implementing a stereo system which incorporates these and the geometric constraints for special surfaces.

Image Description

Research in image description using edges followed the lines of Binford-Horn [Horn 72, Herskovits and Binford 70]. Research went in two directions: first, higher performance systems which extended those techniques; second, simplifications which could be implemented in VLSI. Binford-Horn faced these problems:

1. Areas with large gradients are quite common. Gradient-based edge operators give masses of spurious edges on smooth surfaces where the intensity has large gradients, as on the fuselage of aircraft, or on a stack of blocks where one block illuminates another by reflected light, or at shadows.
2. Edge operators gave poor estimates of locations of edges. The TOPOLOGIST, predecessor to Binford-Horn, and other systems localized edges by finding the maximum of the gradient function, or by thinning bands of points with gradient above threshold. The gradient is flat at maximum, it is a poor function for localization.
3. Edge operators missed low contrast edges. Edge operators must deal with spatially random camera noise and with small surface markings which are signals but are not extended.
4. Edge operators had poor resolution for thin features and at vertices.
5. Edge operators depended on thresholds which were set by experimentation with individual images.

Binford-Horn had an underlying model:

1. Lateral inhibition removes smooth gradients.
2. Zero-crossings of the laterally-inhibited signal gave accurate estimates of locations of edges and their angles.
3. An extended directional operator combined with sequential detection theory provided sensitivity. A non-linear evaluation function improved its robustness in dealing with small markings.
4. Directional operators are appropriate for features which are elongated yet narrow.
5. Thresholds were set by analytic calculation of signal to noise for operators. There were no empirical parameters.

That line of research had succeeded in quantifying performance of edge operators.

Binford-Horn

The Binford-Horn operator proceeded by a raster scan:

a. Obtain the laterally-inhibited signal by subtracting the local average signal (this is equivalent to the Laplacian).

b. Make sums of the gradient in directions about 10 degrees apart.

c. Detect edges from the gradient. If the gradient is large in any direction, then an edge is near.

d. Localize edges in angle and direction from zero crossings of the laterally-inhibited signal. Choose the direction of the maximum gradient and linearly-interpolate the laterally-inhibited signal through zero to estimate transverse position.

e. Localize by suppressing transverse satellites. The laterally-inhibited signal goes to zero not through zero on either side of the edge while the gradient remains large. These satellites were removed by requiring that the odd part of the signal be large compared to the even part.

f. Localize by suppressing longitudinal satellites. This is a problem only with directional operators. In the vicinity of an edge, adjacent points will have a maximum signal by overlapping the edge. Every edge has a halo of satellites. At a point p, the maximum signal passes through point q. If p and q are on an edge, then the local maximum signal at q will pass through p and vice versa. Choose mutual maxima.

g. Extend edges by sequential detection. Test whether this edge element continues adjacent edges by requiring continuity of direction, position, and signal. If so, link with adjacent edge. If not, start a new edge. For each edge, accumulate the sum of Gaussian residuals of edge strengths along the path and over the last few steps of the path.

h. Extend previous edges by extrapolation. In the raster scan, if a continuation of an edge has not been found, make a tentative extension of the edge along the direction of the maximum gradient. The continuation will be tested at every step. new edge. For each edge, accumulate the sum of Gaussian residuals of edge strengths along the path and over the last few steps of the path.

i. At each extrapolation, test the sum of residuals versus a threshold for that path length. If the signal is small enough over the last few steps of the path to be probable with zero signal, terminate the edge. If the total signal over the path is small, eliminate the edge.

j. Determine the edge termination by eliminating steps near the end with signal below the local average of signal along the path.

These operations could be summarized as detection, localization, and linking. We have improved localization operations. We are just about to begin extension of linking operations.

A major problem in edge segmentation remains. Images have features of all sizes, from tiny markings to extended boundaries. Marr has made some prescriptions for combining the results of edge operators of various sizes, but those procedures are adhoc. In order to get improved performance we have begun on this problem and have made some progress, but that research is still underway. The problem appears most evident in textured scenes. In the example of NVL imagery shown below, the stream shows up most clearly because of texture boundaries, i.e. because it is uniform, i.e. has few edges, rather than because its boundaries are clearly defined.

The other aspect of our research has been to implement a simplified edge operator based on the principles described above. In doing so, we have invented an algorithm for region traversal in a raster scan with boundaries interpolated as above. We have tried it on a number of scenes. Results are included at the end of this section. They appear quite useful. We plan to work to include the edge operator in an integrated system.

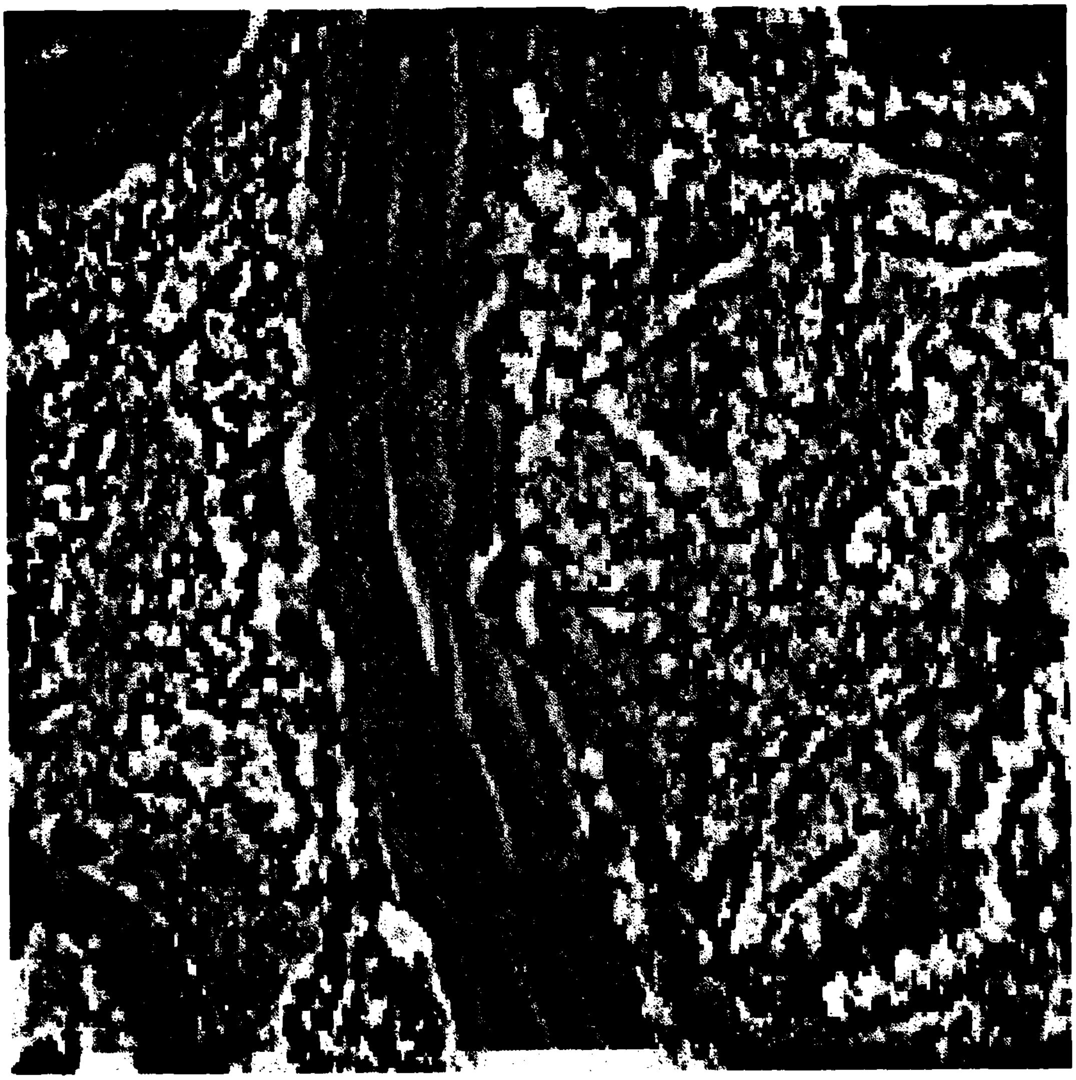


Fig. C-1 Original NVL Image

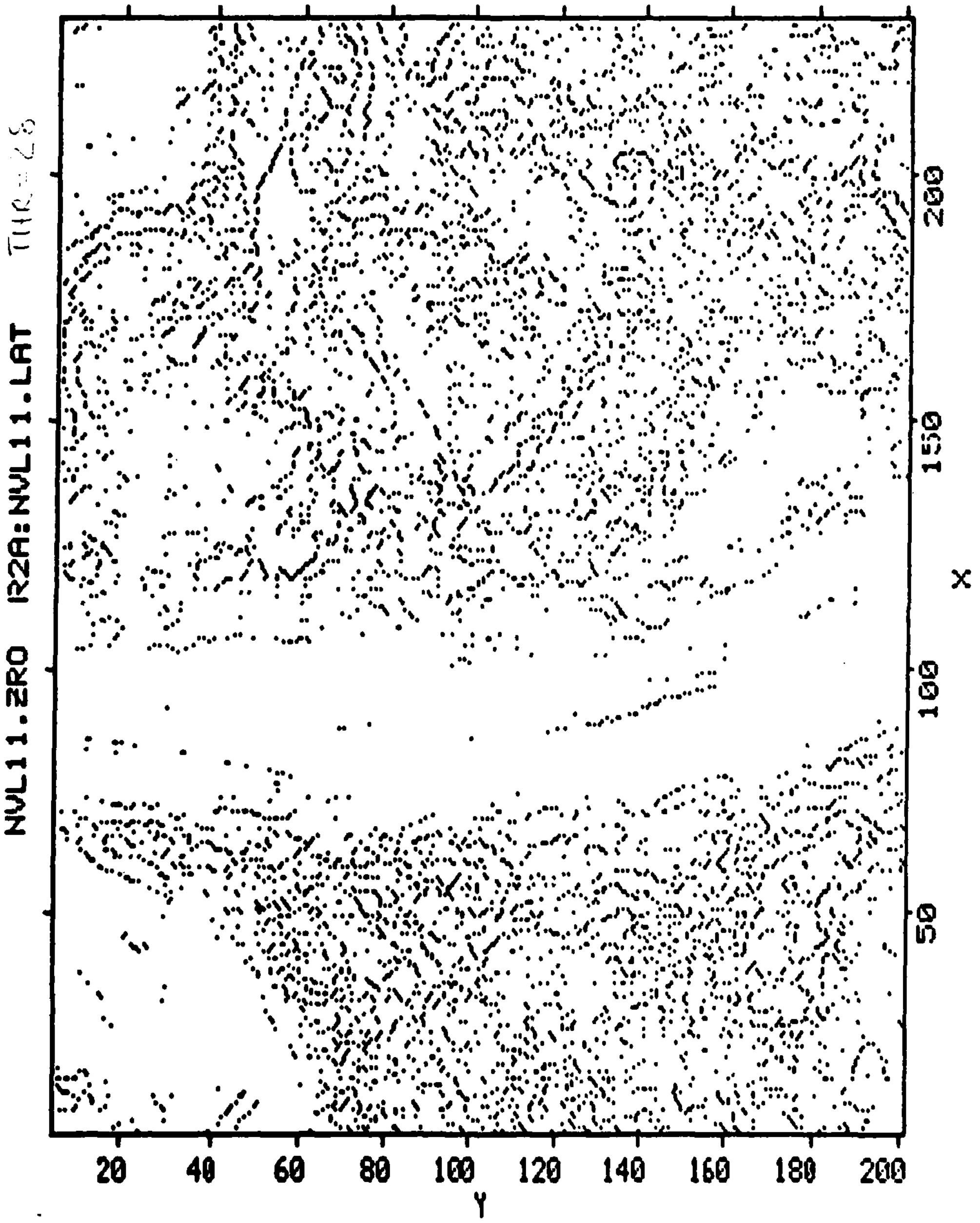


Fig. C-2 Edge Segmentation for NVL Image

[1]

Geometric Constraints for Special Surfaces

This section summarizes research by Liebes in analysis of geometric constraints for special surfaces, namely vertical and horizontal planes and cylinders, applied to the interpretation of monoscopic and stereoscopic imagery of cultural objects. We note that two types of geometrical structural elements abound in cultural objects, orthogonal trihedral vertices, and portions of right circular cylinders. Special case treatment of these elements facilitates quantitative symbolic description of imaged scenes. The approach is based upon the use of projective invariants of edge features. A powerful additional constraint is imposed by the alignment of many kinds of cultural objects with gravity.

Two central goals of a powerful stereo vision system are to make a depth map, i.e. a determination of stereo correspondences, and the construction of a quantitative symbolic description of the depth map. By stereo correspondences we mean a map of disparities, a unique map of features of one image with features of the other image corresponding to the same physical points in the imaged scene. Existing techniques for establishing stereo correspondence perform satisfactorily for smooth terrain, but are inadequate for analyzing topographically complex structures. Even if it were possible to map ranges to arbitrary accuracy and density, there remains the task of making a description of the surfaces of the scene.

While continuous tracking of area correlation of images succeeds with smooth textured terrain, it fails where surface slope and range are discontinuous and for uniform surfaces without textural detail. We search for mechanisms for segmenting surfaces with abrupt changes in range or slope, for segmenting surfaces steeply inclined with respect to the camera, and for inferring object surface structure within textureless regions and within regions visible to only one camera. We are pursuing a dual approach for our image understanding and stereo program that incorporates analysis in both the projective space of single images and in the three dimensional space defined by the camera model.

Vertical and horizontal surfaces, and vertical and horizontal edges occur with such frequency in buildings, storage tanks and pipes in cultural sites that it is valuable to work out special case constraints for these construction elements. We address the important special cases of plane and cylindrical structural elements, especially right parallelepipeds and right circular cylinders aligned with gravity, vertical and horizontal surfaces. Ignoring these capabilities for use of special structures is to throw away valuable information. We have considered initially the case of nadir viewing stereo imagery, and have established the extendability of the nadir case to that of oblique imagery.

We concentrate upon two classes of feature elements that occur frequently in cultural artifacts. The elements are orthogonal trihedral vertices (OTVs) (such as associated with the exterior and interior of box-shaped objects), and portions of right circular cylinders (RCCs). Corners are common elements of buildings, and right circular cylinders are common to storage tanks and pipes. OTVs are cues to cultural structures. We plan to employ them in a collateral development of quantitative shape and orientation information from single images on the one hand, and of powerful stereo correspondence cues and quantitative structural information, on the other.

The approach is based upon the use of projective invariants. Our study of projective transformations and stereoscopic imagery has yielded valuable formulations involving projective invariants, coordinate representations, and stereo edge element organization and analysis. We have applied these formulations to projective invariants of OTVs, namely the locations of the vanishing points for their edges, or equivalently their surface normals. In an application to the important special case of nadir oriented stereo cameras, we have found a simple set of projection and visibility rules that uniquely label the corners. The rules facilitate the quantitative determination, given an OTV in one image, of the appearance of the corresponding OTV in the conjugate image, as a function of relative displacement along the associated epipolar line (the image space projection of a plane passing through the two camera centers and the object space field point).

Consider the case of a canonical nadir viewing stereo camera system, where by "canonical stereo camera system" we refer to two identical cameras with parallel optical axes, and intercamera baseline perpendicular to these axes. The rules utilize for the nadir case the following projective invariant properties: one, all lines that are vertical in object space, thus roughly one-third of the edges associated with buildings, will radiate in projected image space from a single point, the nadir point of the aerial photograph; two, every horizontal surface will project identically in both members of the stereo image pair; three, every object space horizontal right angle will project to the film planes as an identically oriented right angle. These three invariant properties have been incorporated into simple rigorous rules for the prediction and classification of each of the sixteen kinds of interior and exterior corners that can be associated with hollow and solid right parallelepipeds. Projections are uniquely determined by the position of the corner in the image and its orientation about the zenith. These rules allow monoscopic inference of corner configuration, the appearance of conjugate projections of corners as a unique function of position along an epipolar line, and the visibility of faces associated with the corners. These rules offer powerful means of identifying and analyzing buildings, including the capability to analyze microdetails such as doors, windows, chimneys, overhangs, indentations, etc. We have established that these rules extend directly to the general case of arbitrary perspective projections of OTVs, in both the far field and near field cases.

The simplicity of the rule formulation in nadir viewing arises from the facts that the vertical edge vanishing point coincides with the nadir point, and both of the remaining OTV edge vanishing points are oriented at right angles to one another at infinite distance parallel to the film plane. In the more general oblique case, all three vanishing points are at finite distance from one another in the film plane. The rules in the latter circumstance more explicitly utilize the projective relationship of the edges of the sixteen different kinds of corners to the vanishing points. We have demonstrated that elements of the approach extend to the case of vertical cylinders with arbitrary polygonal cross section.

Let us consider now the OTV rules for the case of nadir imaging. Imagine a single building, the facial elements of which are either parallel or to or mutually orthogonal to one another. If such a building were imaged by a nadir viewing camera, the visibility of any given face would depend upon which side of it the projection of the nadir point fell. The development of the rules will be facilitated by considering a RPP wire model, the edges of which are aligned parallel to those of the building. The wire model will serve as a signature rule generator for the corner elements of the building. Let us consider the image space to be divided into four quadrants about the nadir point. All the possibilities for placement of the variously oriented vertical faces relative to the nadir point can be accounted for by considering a separate wire model in each of the four quadrants, the quadrant boundaries being aligned with the horizontal edges of the models.

The arrangement of the four wire models is illustrated in Fig. 1. The flight line, or direction of the intercamera baseline, extends left to right in the figure. Quadrant 1 is defined to be the first full quadrant encountered rotating counterclockwise from the right extension of the flight line from the nadir point. A corner labeling convention is indicated, unprimed labels being associated with the top of the model and primed labels with the bottom. We are about to link corner labels with corner signatures, the latter relating to the directions of the corner edges relative to those of the quadrant boundaries. For this reason the corner labeling in any application must be in consistent relationship to the quadrant boundary labeling. The directions of the four quadrant boundaries are labeled "E" ("east"), "N" ("north"), "W" ("west") and "S" ("south"), where the "compass" points are relative rather than geographic. "East" is defined to be directed away from the nadir point along the boundary between quadrants 1 and 4. Figure 2 introduces a corner-signature scheme. The scheme involves indicating for each corner the directions of the edges associated with the corner. For example, wire model corner A has associated with it edges directed W, S, and T, where T indicates an edge directed toward the the nadir point (A would indicate an edge directed away from the nadir point). We can say that *corner A has signature* WST. It will furthermore be observed that the signature for wire model corner A is invariant under lateral translation, that is the signature is independent of the quadrant within which it resides. It will furthermore be observed that the signature of each of the wire model corners is unique.

Immediately following the listing of the eight wire model corner-signatures in Fig. 2, we list the corner-signatures for a quadrant 1 solid (S). (The "solid" terminology was initially introduced to distinguish from "wire model". Subsequently, rectangular holes were introduced into the solid. The interior corners of the holes are just as much associated with the solid object as are the exterior corners. The use of the nomenclature "solid" to refer to solid exterior corners is no longer apt, and though persisting into this report will subsequently be revised.) The signatures for each of the solid corners is a obtained by masking the signatures of the wire model corners according to the visibility of the associated edges. For example, solid corner A bears signature WS, compared to the corresponding *wire model corner signature* WST, since for the solid the edge directed toward the nadir point is not visible in this quadrant. Figure 2 additionally develops, for a block in quadrant 1, the signatures for the corners of rectangular holes appearing in the top (TH), sides (SH, WH, NH, EH), and bottom (BH) of the block. It will be seen that there is a uniqueness in the corner labeling, with the exception of the degeneracy of the signatures of the A' corners in the holes. It is also clear that the use of edge direction information alone is insufficient to distinguish between a hole that bottoms out and one that penetrates the far side.

The corner signatures for blocks and holes situated in other quadrants are developed in like fashion to that indicated here for quadrant 1. It can be seen from rotational symmetry considerations that an analogous uniqueness of corner signature assignments will apply for the other quadrants. Thus, whereas the signatures of corners that differ only by their translational position within the field of view can vary from quadrant to quadrant, the combination of signature and quadrant assignment is sufficient to uniquely label the corner type.

Figure 3 illustrates a stereo view of several examples of structures to which the above discussed labeling and corner-signature scheme has been applied. For convenience in the example, all the structures are illustrated in a common state of rotational alignment. Thus the quadrant "compass" directions associated with each structure are the same. Had the structures not been similarly aligned, a different set of quadrant boundaries would apply to each structure. The structures appear in various quadrants throughout the figure. The corners have been labeled according the procedures just discussed, where for each corner, a signature and quadrant has been assigned, and from it a label determined. Since the stereo pair of images are related by a simple lateral translation of camera through object space, the labeling scheme yields like labels for corresponding corners in the two images. A leading "S" on a label donotes "solid" (solid exterior corner), a "TH" denotes "top hole", "SH" south hole, etc. We do not address here the issue of distinguishing between solid and wire structures, and bottomed-out vs. penetrating holes.

One of the real problems in unravelling stereo correspondences in complex scenes, particularly those acquired under high conververgence angle conditions (base line an appreciable fraction of range), is the difficulty of sorting out the ambiguities among candidate match points. There is generally a trade off between having to contend with multiple ambiguities at the microlevel and structural complexity at the macrolevel. OTV labeling is a device for capitalizing on the high information content of unique regions at the quasi-macrolevel.

It is planned to construct OTV corner finder operators. The number of corners found will be a very small fraction of the number of resolvable points in the scene, and even a relatively small fraction of the number of resolution elements associated with edges. It is therefore planned early in the processing to search for correspondences among corners. Any correspondence ambiguities that do persist along epipolar lines will very likely be resolved by recourse to the constraint that corners corresponding to a common physical feature must bear the same corner label. We intend to use the 3-dimensional corner orientation information derived from the application of the corner finding and labeling operations as a powerful guide in both the monoscopic and stereoscopic derivation of a quantitative symbolic description of the scene. The establishment of corner labels and corner stereo correspondences can be used to infer the object space location of edges connecting corners, and thereby to predict and help unravel ambiguities among conjugate edge candidates and to infer surface form between edges. Additionally, it will enable the inference of structure within regions where only monoscopic imagery exists due to obscuration in one member of the stereo pair. These constraints on surface shape will enable more complete and accurate measurement of dimensions where they apply.

We do not describe here the extension that we are undertaking from the nadir viewing geometry to oblique near-field stereo. The concepts are analogous, but entail an added level of geometric complexity that derives from the fact that all three of the vanishing points of the edges of the OTVs are now generally at finite distance from one another on the image plane.

We are also developing a related approach to dealing with right circular cylinders. This part of the study has not yet been given the level of attention that has been accorded to the OTVs. It is our intent to undertake a related extension of the concepts discussed in this section to right cylinders of arbitrary cross section.

We expect that insights developed in the further development and implementation of the work described here will guide us to substantially more powerful mechanisms for automated image understanding.

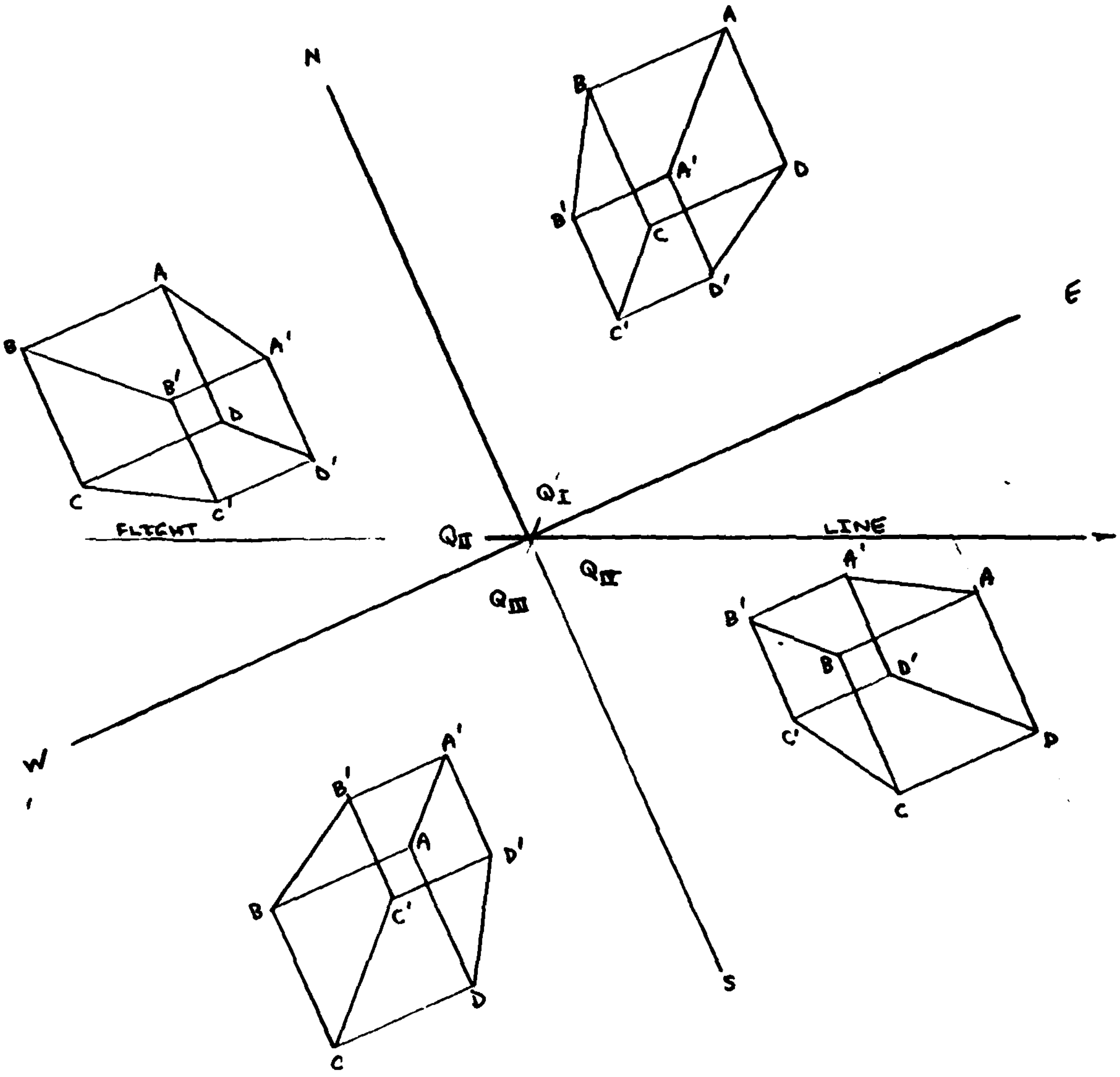
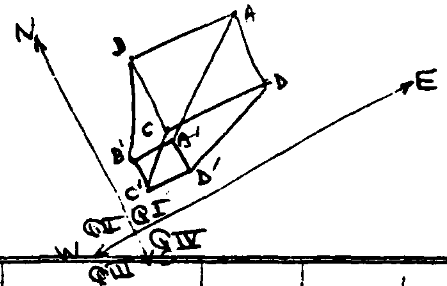


Fig. 1 Four Wire Frame Models



		N	W	S	E	A	T	
WIRE	A		X	X			X	
	B			X	X		X	
	C	X					X	
	D	X	X				X	
	A'		X	X		X		
	B'			X	X	X		
	C'	X			X	X		
	D'	X	X			X		
SOLID(S)	A		X	X				
QI	B			X	X		X	
	C	X			X		X	
	D	X	X				X	
	A'							
	B'			X		X		
	C'	X			X	X		
	D'		X			X		
TOP HOLE(TH)	A		X	X			X	
QI	B			X	X			
	C	X			X			
	D	X	X					
	A'	(	X	X		X	)	
	B'							
	C'							
	D'							
SOUTH HOLE(SH)	A							
QI	B							
	C				X		X	
	D		X				X	
	A'	(	X	X		X	)	
	B'							
	C'				X	X		
	D'	X	X			X		
WEST HOLE(WH)	A							
QI	B			X			X	
	C	X					X	
	D							
	A'	(	X	X		X	)	
	B'			X	X	X		
	C'	X				X		
	D'							

Fig. 2 Corner Signature Scheme

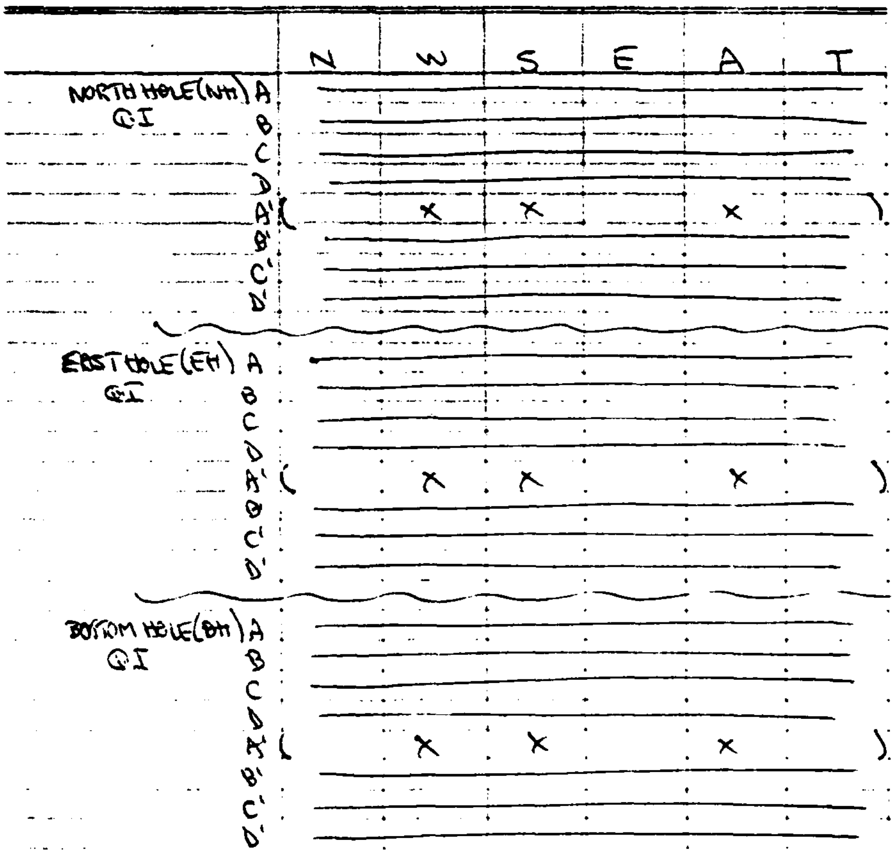

N
W
S
E
A
T
NORTH HOLE (NH)
@ I
EAST HOLE (EH)
@ I
BOTTOM HOLE (BH)
@ I
A
B
C
D
A'
B'
C'
D'
x
x
x

Fig. 2 (Cont.)

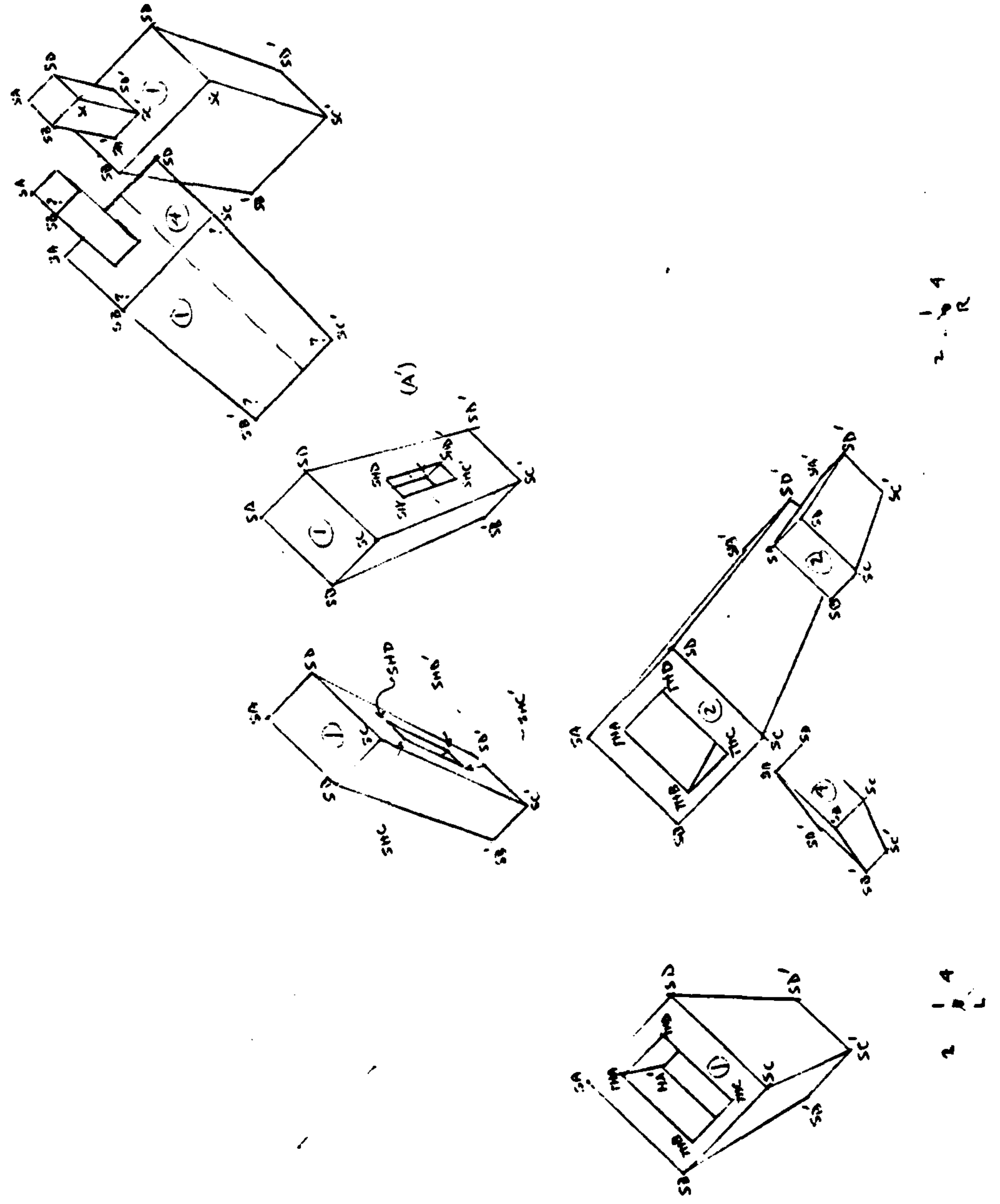


Fig. 3 Stereo View of Structures

Geometric constraints in stereo vision

R.D. Arnold and T.O. Binford

Computer Science Department, Stanford University
Stanford, California, 94305

Abstract

The correspondence problem, matching the same feature in two views, is a central problem of stereo vision. We examine geometric constraints on stereo correspondence and describe progress toward formulation from first principles of an evaluation function for selection of the best among alternative correspondences. Within a surface interpretation, conditions on correspondence of edges and surface intervals are shown. These conditions are useful with wide angle stereo and provide particularly tight constraints for narrow angle stereo. We invoke the general assumption that edge and surface orientations are not related to observer position. We are combining these general constraints with ongoing work in scene modeling of known regularities which include distributions with respect to the gravity vector (horizontal and vertical edges and surfaces), parallelism and alignment with local coordinate systems, and orthogonal corners. We will use them to calculate a likelihood measure for correspondences.

Introduction

The fundamental task in the stereo problem is to establish a correspondence between features or regions in two or more images from which we can calculate positions in three dimensions. Traditionally, these correspondences have been based on correlation of areas, but recent attempts have been made here to calculate stereo correspondences on the basis of edges[1]. Control Data Corporation has used a technique for cultural scenes which combines edge segmentation with cross correlation of intensities between edges[3,5]. The MIT approach achieves a somewhat similar effect by matching zero crossings, including those at high resolution[2,4]. The use of features such as edges has the important advantage of working in scenes with spatial discontinuities, which are typical of cultural objects. For example, a scene may contain the image of a roof adjacent to a parking lot below. In the other view, the same patch of roof may be adjacent to a different patch of parking lot. An area-based correlation overlapping this discontinuity would find a poor match, while an edge-matching stereo system could identify the roof edge accurately in both views. The goal of our research is to combine the useful characteristics of feature-based stereo with area-based stereo. However, our approach is to discover how much information can be derived from each source independently.

This paper addresses the problem of correspondences based on minimal edge information. The edge operators we use supply information on brightness and contrast as well as position and orientation. However, image intensity can be affected by several factors. Film and camera sensitivity may vary. Reflectivity may depend on viewer position as in specular reflections. Finally, surface character and illumination may change for images taken at different times. Edge position and orientation are much more stable quantities because the intensity changes listed above will not significantly affect them. These are the only parameters considered in this paper.

We also make the general assumption that the scene is independent of the viewer. While the stereo camera model and the objects in the scene may be aligned with respect to some common reference, such as gravity, individual features in the scene should have no dependence on camera angle. That is, small shifts in camera position should not cause significant changes in the image.

Stereo geometry

We will use the stereo camera geometry of figure 1. The origin is located at the focus of the left camera and the right camera focus lies on the x axis. The two image planes are coplanar and are perpendicular to the z axis. The baseline, B, is the distance between focii, and we will use the image distance, f, as the unit of length. I.e. , $f = 1$. This is our "normal" camera model. If an actual stereo image pair were taken with a different geometry, we could use a few simple transformations on the images to produce a pair consistent with this model.

Given any point on an object, we can define an "epipolar plane" as that plane passing through the object point and both focii. This plane intersects the two image planes, defining an "epipolar line" in each. These lines are parallel to the x axis in our normal camera model, and

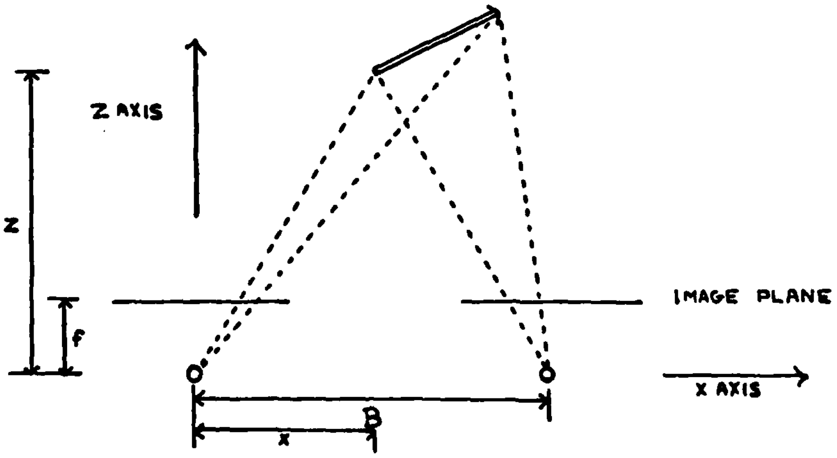


Figure 1. Normal camera model.

we will refer to them as "epipolars". Corresponding points must lie on corresponding epipolars, that is lines with the same y-coordinate in both left and right images.

In the discussion that follows, we will be concerned with matching features in a given left epipolar with corresponding features in the right epipolar line. The feature parameters of interest are the position and orientation of edges, that is, the points at which image edges intersect the epipolar, and the slopes of the edges at those points.

Edge angles

Given a corresponding pair of edges, one in the left, one in the right, we are interested in how their angles are related. In general, the two angles may take on any values, but intuitively it seems they should usually be similar, especially for moderate or small baselines. This is in fact the case, as we will now show.

Consider an object edge passing through a point (x, y, z). The edge has an orientation in three dimensions, and can be characterised as a point on the surface of a unit sphere, whose origin is (x, y, z). This is known as the "gaussian sphere," and points are located on its surface in terms of spherical coordinates θ and φ. See figure 2. The spherical coordinate axis is parallel to the z axis and θ corresponds to longitude, measured counter-clockwise from the x axis when viewed from the cameras. φ corresponds to latitude and is measured from the sphere's axis.

The edge projects to a line intersecting the epipolar lines defined by (x, y, z). The angle of the image line is θ_l in the left image and θ_r in the right image, measured counter-clockwise from the x axis. A continuous function maps points on the gaussian sphere to pairs of image angles, (θ_l, θ_r). Similarly, there is an inverse function which maps points in the space $\theta_l \times \theta_r$ to points on the gaussian sphere. This inverse function is defined everywhere except at $(0, 0)$. This is because the great circle of points on the sphere for which $\theta = 0$ all map to $(0, 0)$, and the function is not invertible at that point.

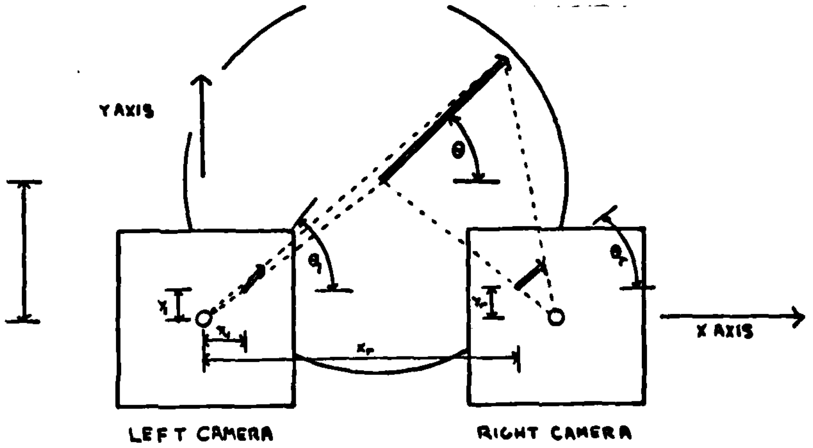


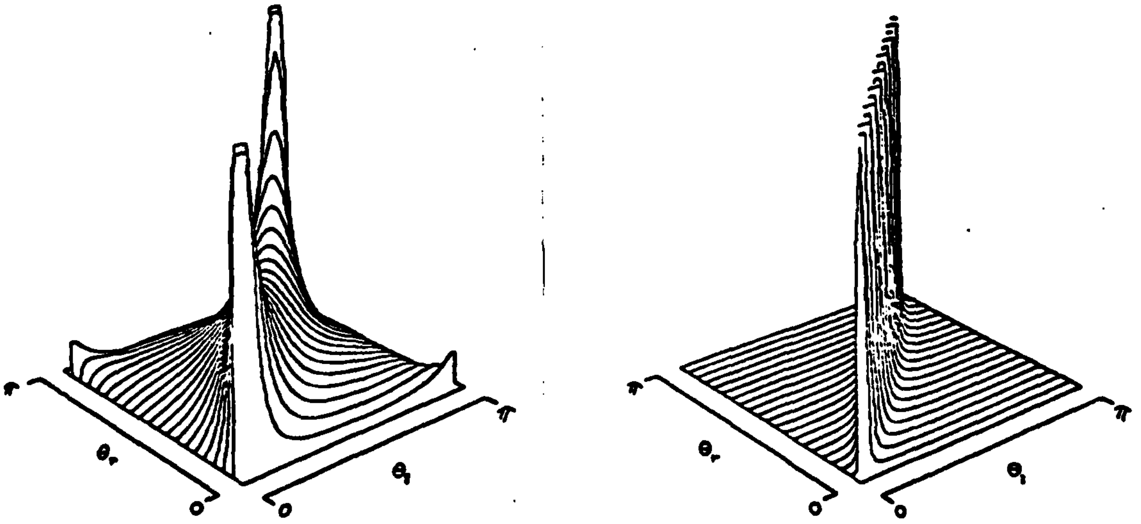
Figure 2. Gaussian sphere for edge angle derivation.

Given this mapping, we are now able to translate probability distributions in one domain to probability distributions in the other. For example, we are interested in the *following problem:* Assume a uniform distribution on the gaussian sphere. What distribution of image angles is expected? In other words, if all object edges are randomly and uniformly distributed in orientation, are some combinations of (θ_l, θ_r) more likely than others?

We know the mapping from $\theta_l \times \theta_r$ to $\theta \times \varphi$. The determinant of the matrix of partial derivatives (Jacobian matrix) is the scale factor for area under the mapping, and thus is the scale factor for probability density. Suppose point (a, b) in $\theta_l \times \theta_r$ maps to (A, B) in $\theta \times \varphi$, and that the determinant of the Jacobian at (a, b) is D. Then a small patch around (a, b) maps to a patch around (A, B) with D times the area. If the probability density at (A, B) is P, then the probability density at (a, b) is DP.

Figure 3 shows the function D plotted for a stereo baseline typical of aerial photographs, $B/z = 0.7$. For the uniform distribution assumption, this surface corresponds directly to probability distribution over $\theta_l \times \theta_r$. The surface forms a high, narrow saddle along the line $\theta_l = \theta_r$, with a singularity at (0,0). This corresponds to the intuitive notion that left and right angles are usually similar, but the sharpness is surprising. Half width at half maximum (HWHM) at the center is 30 deg.

As figure 4 shows, probability functions for narrower baselines are even sharper. $B/z = 0.07$ corresponds to human vision at a range of about 1 m, with a HWHM at the center of 3 degrees. For a human, about 9 degrees HWHM would be adequate to fuse stereo images at 1 foot. We conclude that in biological vision: a) stereo cells should show a half width of about 9 degrees for angle differences in the two eyes; b) such stereo cells will be insensitive to angles of vectors in



Figures 3,4. Probability density as a function of θ_l and θ_r. $B/z = 0.7$ on the left and 0.07 on the right. Both graphs have $x/z = .5B/z$, and the peaks are truncated.

space. However, those angles can be calculated accurately at a later stage from assocated vectors. We have not seen this observation in the literature, but experiment supports this interpretation. Nelson, Kato, and Bishop[6] show stereo cells with orientation half widths from 10 degrees to 20 degrees, which are insensitive to space angles of vectors.

Another way of looking at the data is to consider the distribution of "wrong matches". Suppose we choose an edge from the left and an edge from the right at random, and try to interpret them as corresponding. This will produce a distribution of edges in 3 dimensions, i. e. on the surface of the gaussian sphere. The nature of the distribution will depend on the original distributions of θ_l and θ_r. For the case of a uniform gaussian sphere distribution, it is easy to show that θ_l and θ_r are also uniform. For each value of θ_l in the image, there is a corresponding set of points on the gaussian sphere. This set of points forms a great circle, that is a circle of unit radius. The probability of a particular value of θ_l occuring depends on the integral of the gaussian sphere probability distribution over that circle. If we assume a uniform distribution on the sphere, then all circles will yield identical integrals. Similarly, θ_r will be uniformly distributed. Figure 5 shows a distribution for $B/z = 0.7$, under the assumption of uniform distribution. The distribution, which is actually on the surface of the sphere, has been cut in half and projected onto the plane of the image for display. The result is a sharply double peaked distribution, with each peak oriented toward (and the missing half away from) a camera. This violates the assumption that the scene should be independent of the observer, and such a distribution could be used to identify wrong matches.

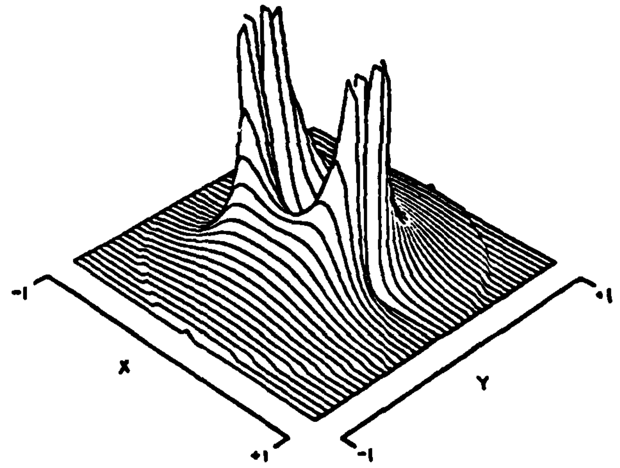


Figure 5. Plot of "wrong match" distribution versus θ and φ, projected onto a plane parallel to the image planes. Camera parameters are same as for figure 3, and peaks are truncated.

Edge intervals

Given an object surface, its image at a particular epipolar line will generally consist of two bounding edges and the interval between them. Thus we can talk about *corresponding intervals* and ask how their lengths are related. In general, the lengths can take on any values, but again our intuition is that for moderate or small baselines, they should usually be comparable.

As discussed above, choosing a particular epipolar line defines an epipolar plane. Assume that this plane cuts an image surface, forming a continuous profile. Now consider the case where the profile consists of a central small segment flanked by two larger ones extending off to the left and right. (Figure 6). Assume the vertices between the segments are edges which can be located in the images. We want to vary the orientation of the small segment and see what happens to its image. In general, the left and right images will show an interval between two edges. The length of the interval will depend on the orientation of the segment and its position with respect to the cameras. For some orientations, one of the edges may be occluded and the segment itself not visible.

We see immediately that there is a *simple function* mapping orientation, ϑ, to projected interval lengths, p_l and p_r. What is needed is an inverse function mapping some image parameter to ϑ. To do this, we define a ratio, $R = p_r/p_l$. This has the advantage of reducing the information from the image lengths into a single number while eliminating the dependency on segment length, d. Now we can easily invert the function and, by analogy with the angles discussion, take the derivative. Then $d\vartheta/dR$ is a scale factor which indicates how much a unit length in "R-space" is stretched in mapping to "ϑ-space". This allows us to translate probability densities as before. For example, suppose an interval ratio a maps to an orientation A, the derivative of the mapping at a is D, and the probability density at orientation A is P. Then the probability density for ratio a is DP.

This derivative D is normalised and plotted against R in figure 7. ϑ ranges over 180 degrees

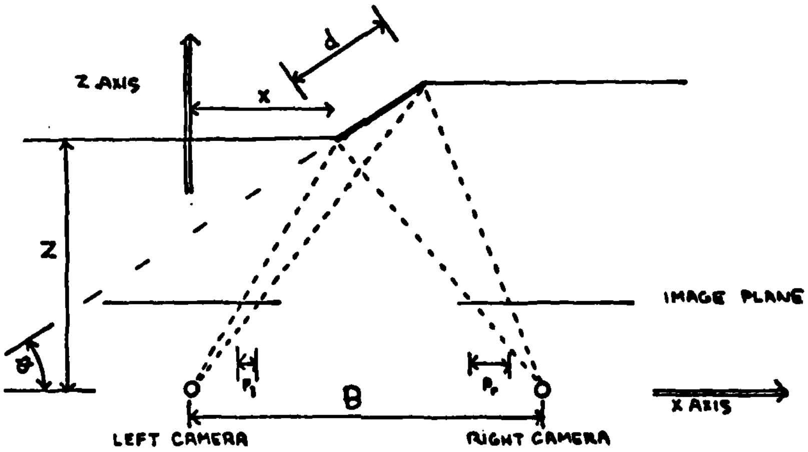


Figure 6. Geometry for edge interval derivation

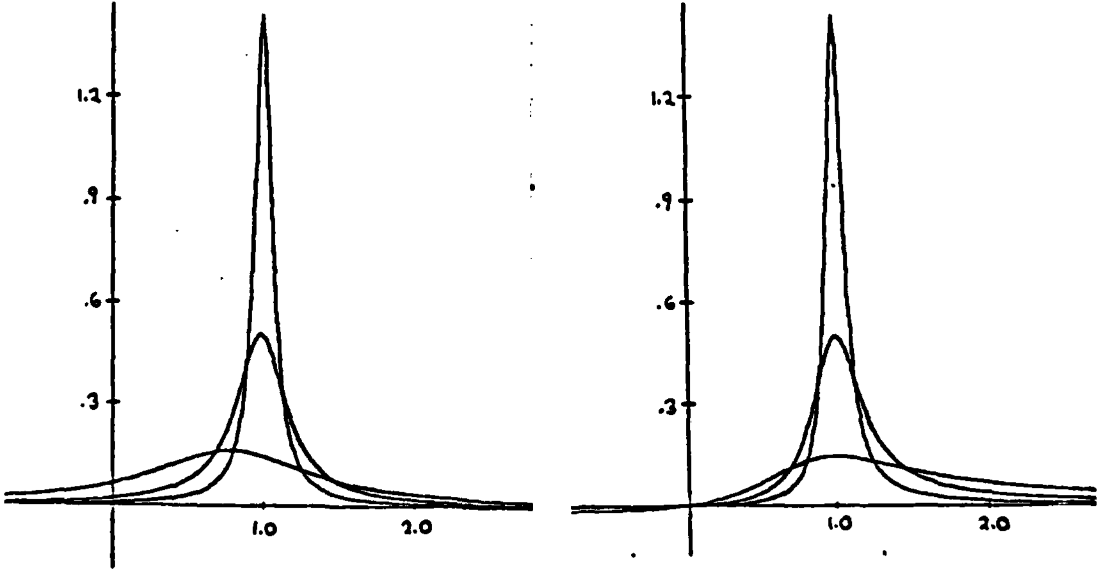
while the domain R extends from $-\infty$ to $+\infty$. A ratio of sero corresponds to a surface exactly in line with the right camera, while a ratio of $\pm\infty$ corresponds to alignment with the left camera. Negative ratios result when the surface presents a different face to each camera. Assuming the surfaces are opaque, the small segment is visible to only one camera and this corresponds to occlusion.

If we assume that the small segment takes on all orientations uniformly, then D exactly equals the probability density for interval ratios. The peak near $R = 1$ indicates that most intervals tend to have comparable lengths and is much more noticeable for narrower baselines. See figure 7. Half of all ratios for $B/z = 0.07$ (human stereo at 1 m) lie between 0.93 and 1.07. Note that integrating the probability function between $-\infty$ and 0 gives the range of angles for which occlusion may occur, which when normalised is the probability of occlusion.

There is one problem with using the results of figure 7 directly. While the function gives the true probability density per unit R, dR is a nonuniform unit varying in length from 0 to ∞. Consequently, orientations which are simple reflections, i. e. ϑ and $-\vartheta$, yield different values. To adjust for this, we scale the derivative by a factor of R, yielding the function in figure 8. This satisfies the conditions of symmetry, in that symmetric orientations now have identical values. Another way to get this same result is to use $\log R$ as the image parameter and take the derivative of ϑ with respect to $\log R$.

Conclusion

The two functions calculated above are actually scaling functions which allow a distribution function in the object space to be translated to a distribution function in the image space. The



Figures 7,8. Probability density as a function of interval ratio. For both figures, $B/z = 0.7, 0.2$ and 0.07; $x/z = .5B/z$. The left graph is the original function, the right graph is scaled by R.

functions are sharply peaked even for the moderate baselines used in aerial photography. When baselines corresponding to human vision are used, the conditions are extremely strong. This leads us to reccommend the consideration of using mapping sequences with small angle stereo. For example, instead of a pair of images with a 60 degree baseline, a sequence of 10 images at 6 degrees would provide the same overall baseline. Tight constraints would simplify the task for the program, which could track features from frame to frame, then make depth estimates based on the accumulated baseline.

We have been using the results of these functions as components of an evaluation function for stereo matching. The problem is to distinguish among the many physically possible matches those that are most likely. As we noted, many factors enter into such a judgement, but the two in this paper are among the most basic, requiring little *a priori* knowledge of the scene. Applying the evaluation function to sample pairs of epipolar lines has resulted in the most plausible stereo interpretations receiving the highest scores. The calculations give quantitative support to our intuition that left and right images will be similar.

To first order, the assumption of uniform distributions in the object space is useful. However, the functions can easily incorporate knowledge of the scene when it is available. In particular, cultural scenes tend to be strongly oriented with respect to gravity for obvious structural reasons. This information could be represented by a distribution function for edge orientations that has a strong peak for vertical edges and a narrow band for horizontal surfaces. We will be estimating such distributions in future work and expect improved results with them.

References

[1] Arnold, R.D., "Local Context in Matching Edges for Stereo Vision," *Proc. Image Understanding Workshop*, Boston, May 1978.

[2] Grimson, W.E.L., and D. Marr, "A Computer Implementation of a Theory of Human Stereo Vision," *Proc. Image Understanding Workshop*, Palo Alto, April 1979.

[3] Henderson, R.L., W.J. Miller and C.B. Grosch, "Automatic Stereo Reconstuction of Man-Made Targets," *Proc. SPIE*, Huntsville, August 1979.

[4] Marr, D. and T. Poggio, "A Theory of Human Stereo Vision," *AI Memo 451*, MIT, November 1977.

[5] Mori, K., M. Kidode and H. Asada, "An Iterative Prediction and Correction Method for Automatic Stereo Comparison," *Computer Graphics and Image Processing*, 2,393, 1973.

[6] Nelson, J.I., H. Kato and P.O. Bishop, "Discrimination of orientation and position disparities by binocularly activated neurons in cat striate cortex," *Journal of Neurophysiology*, 40(2):260–283, March 1977.

Derivation, angles

We wish to derive the function mapping $\theta_l \times \theta_r \mapsto \theta \times \varphi$, where

$$0 \leq \theta_l, \theta_r < \pi$$
$$0 \leq \theta < \pi$$
$$0 \leq \varphi \leq \pi.$$

The approach is to convert to rectangular coordinates, do the stereo projections, and convert to spherical coordinates. The stereo projections are given by

$$(x_l, y_l, z_l) = \left(\frac{f}{z}x, \frac{f}{z}y, f\right)$$
$$(x_r, y_r, z_r) = \left(\frac{f}{z}(x - B), \frac{f}{z}y, f\right).$$

The inverse projections are given by

$$z = \frac{fB}{x_l - x_r}$$
$$y = \frac{z}{f}y_l = \frac{z}{f}y_r$$
$$x = \frac{z}{f}x_l = \frac{z}{f}x_r + B,$$

where f is the image length, B is the base line, (x, y, z) is a point on the object, (x_l, y_l, z_l) is a point in the left image, and (x_r, y_r, z_r) is a point in the right image.

Now consider a unit vector in the left image, centered at (x_l, y_l), at angle θ_l. The tip of the vector has coordinates

$$x_l' = x_l + \cos\theta_l$$
$$y_l' = y_l + \sin\theta_l.$$

The corresponding point in the right image will have the same y-coordinate, thus its length must be $\sin\theta_l / \sin\theta_r$, and

$$x_r' = x_r + \frac{\sin\theta_l}{\tan\theta_r}$$
$$y_r' = y_r + \sin\theta_l,$$

where θ_l is the angle in the left image plane and θ_r is the angle in the right image plane.

We now inverse-project to get the points (x, y, z) and (x', y', z') in object space, the origin and tip of the vector respectively. Note that this vector will not have unit length, but will supply the correct value for θ and φ. The values $x' - x$, $y' - y$, and $z' - z$ will be needed:

$$
\begin{aligned}
x' - x &= \frac{z'}{f}(x_l + \cos\theta_l) - \frac{z}{f}x_l \\
&= B\left(\frac{x_l + \cos\theta_l}{x_l - x_r + \cos\theta_l - \frac{\sin\theta_l}{\tan\theta_r}} - \frac{x_l}{x_l - x_r}\right) \\
y' - y &= B\left(\frac{y_l + \sin\theta_l}{x_l - x_r + \cos\theta_l - \frac{\sin\theta_l}{\tan\theta_r}} - \frac{y_l}{x_l - x_r}\right) \\
z' - z &= fB\left(\frac{1}{x_l - x_r + \cos\theta_l - \frac{\sin\theta_l}{\tan\theta_r}} - \frac{1}{x_l - x_r}\right).
\end{aligned}
$$

To simplify further calculations, use the following substitutions:

$$
\begin{aligned}
Q &= \frac{B}{(x_l - x_r)(x_l - x_r + \cos\theta_l - \frac{\sin\theta_l}{\tan\theta_r})} \\
U &= (x_l - x_r)\cos\theta_l - x_l\left(\cos\theta_l - \frac{\sin\theta_l}{\tan\theta_r}\right) \\
V &= (x_l - x_r)\sin\theta_l - y_l\left(\cos\theta_l - \frac{\sin\theta_l}{\tan\theta_r}\right) \\
W &= -f\left(\cos\theta_l - \frac{\sin\theta_l}{\tan\theta_r}\right).
\end{aligned}
$$

Then $x' - x = QU$, $y' - y = QV$, and $z' - z = QW$ and we can easily convert to spherical coordinates:

$$
\begin{aligned}
\theta &= \tan^{-1}\left(\frac{y' - y}{x' - x}\right) = \tan^{-1}\left(\frac{V}{U}\right) \\
\varphi &= \cos^{-1}\left(\frac{z' - z}{\sqrt{(x' - x)^2 + (y' - y)^2 + (z' - z)^2}}\right) \\
&= \cos -1\left(\frac{W}{\sqrt{U^2 + V^2 + W^2}}\right).
\end{aligned}
$$

The Jacobian matrix is:

$$
J = \begin{pmatrix} \frac{\partial\theta}{\partial\theta_l} & \frac{\partial\theta}{\partial\theta_r} \\ \frac{\partial\varphi}{\partial\theta_l} & \frac{\partial\varphi}{\partial\theta_r} \end{pmatrix}.
$$

To calculate these values, we will need the partial derivatives of U, V, and W:

$$\begin{aligned}
\frac{\partial U}{\partial \theta_l} &= -(x_l - x_r)\sin\theta_l + x_l\left(\sin\theta_l + \frac{\cos\theta_l}{\tan\theta_r}\right)\\
\frac{\partial V}{\partial \theta_l} &= (x_l - x_r)\cos\theta_l + y_l\left(\sin\theta_l + \frac{\cos\theta_l}{\tan\theta_r}\right)\\
\frac{\partial W}{\partial \theta_l} &= f\left(\sin\theta_l + \frac{\cos\theta_l}{\tan\theta_r}\right)\\
\frac{\partial U}{\partial \theta_r} &= \frac{-x_l\sin\theta_l}{\sin^2\theta_r}\\
\frac{\partial V}{\partial \theta_r} &= \frac{-y_l\sin\theta_l}{\sin^2\theta_r}\\
\frac{\partial W}{\partial \theta_r} &= \frac{-f\sin\theta_l}{\sin^2\theta_r}.
\end{aligned}$$

Now we have

$$\frac{\partial \theta}{\partial x} = \frac{\partial}{\partial x}\tan^{-1}\frac{V}{U} = \frac{U\frac{\partial V}{\partial x} - V\frac{\partial U}{\partial x}}{U^2+V^2}.$$

Substituting $d = \sqrt{U^2+V^2+W^2}$, we have

$$\begin{aligned}
\frac{\partial d}{\partial X} &= \frac{U\frac{\partial U}{\partial X} + V\frac{\partial V}{\partial X} + W\frac{\partial W}{\partial X}}{\sqrt{U^2+V^2+W^2}}\\
\frac{\partial \varphi}{\partial X} &= \frac{\partial}{\partial X}\cos^{-1}\frac{W}{d} = \frac{W\frac{\partial d}{\partial X} - d\frac{\partial W}{\partial X}}{d\sqrt{d^2-W^2}}\\
&= \frac{W(U\frac{\partial U}{\partial X} + V\frac{\partial V}{\partial X} + W\frac{\partial W}{\partial X}) - \frac{\partial W}{\partial X}(U^2+V^2+W^2)}{(U^2+V^2+W^2)\sqrt{U^2+V^2}}.
\end{aligned}$$

We can calculate the components of the Jacobian matrix by substituting θ_l or θ_r for X. The determinant is then

$$\det J = \frac{\partial\theta}{\partial\theta_l}\frac{\partial\varphi}{\partial\theta_r} - \frac{\partial\theta}{\partial\theta_r}\frac{\partial\varphi}{\partial\theta_l}.$$

Finally, the scale factor for area is given by correcting for the area distortion of the spherical coordinates.

$$\text{area } = \det J \sin\varphi$$

Derivation, intervals

Referring to figure 6, we assume that x, z, B, d, and ϑ are given. The projected interval lengths, p_l and p_r are determined:

$$\frac{p_l + x}{d\cos\vartheta + x} = \frac{z}{z + d\sin\vartheta}$$

$$p_l = d\frac{z\cos\vartheta - x\sin\vartheta}{z + d\sin\vartheta}$$

$$\frac{B - p_r + x}{B - d\cos\vartheta + x} = \frac{z}{z + d\sin\vartheta}$$

$$p_r = d\frac{z\cos\vartheta + (B - x)\sin\vartheta}{z + d\sin\vartheta}.$$

Letting $R = p_r/p_l$,

$$\begin{aligned} R &= \frac{z\cos\vartheta + B\sin\vartheta - x\sin\vartheta}{z\cos\vartheta - x\sin\vartheta} \\ &= 1 + \frac{B/z}{\cot\vartheta - x/z}. \end{aligned}$$

Now we need ϑ as a function of R. Letting $a = B/z$ and $b = x/z$,

$$\vartheta = \tan^{-1}\frac{1}{\frac{a}{R-1} + b}.$$

The density scale factor will be given by:

$$\frac{d\vartheta}{dR} = \frac{a}{(a + b(R-1))^2 + (R-1)^2}.$$

EDGE BASED STEREO CORRELATION

H. Harlyn Baker

Artificial Intelligence Laboratory, Computer Science Department
Stanford University, Stanford, California 94305

Abstract

An edge based approach to stereo depth measurement is described. Its processing consists of extracting edge descriptions of a pair of images, linking these edges to their nearest (projective) neighbors to obtain the connectivity structure of the images, correlating the edge descriptions in an epipolar [3] reference frame on the basis of local edge properties (here, assuming the input images are registered such that scanlines correspond), and cooperatively removing those edge associations formed by the correlation which violate the connectivity structure of the two images.

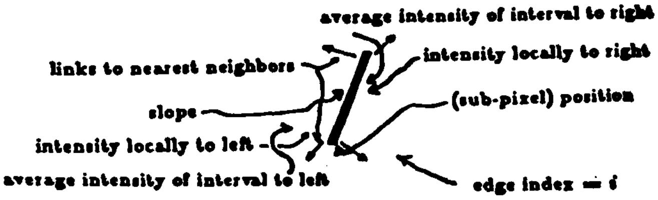
Edge Correlation

The long term interest of this research is *in enabling a computer to build 3-dimensional models of the components of its environment.* To build such models one must have an automatic process for obtaining three-dimensional information from a scene. This is the immediate aim of the work described here - *to develop a vision system capable of obtaining an accurate and reliable edge based depth map of a scene from stereo pairs of views of it.*

Accurate and reliable determinations of the sort needed here require exploitation of the semantic redundancy in the information available to the sensors. The approach to be outlined uses this - intermixing *local* and *global* constraints, constraints derived from observations on the imaging process, in seeking a 3-dimensional interpretation. A correlation procedure chooses the best correspondence of the images using local constraints on a scanline-by-scanline basis, and a cooperative consistency enforcement process works to assure 3-space connectivity using the global constraint of projective connectivity.

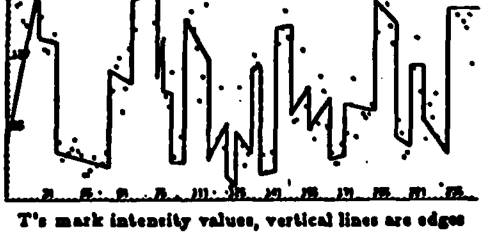
The 3-D correlation was chosen to be *edge* based. This is because of the higher accuracy associated with edge positioning than with area matching, the reduced computation and combinatorics in dealing with edges rather than with area templates, and the desire (at least initially) to work with those parts of the image (and hence of the scene) with the greatest information content - the locus of intensity contrasts between surfaces.

Edges, as they are defined here, occur at those places in an image where the second derivative from a 7×1 or 1×7 bar operator (with lateral inhibition) crosses zero. Each edge has a two dimensional *slope* in the image plane (only those edges with a vertical component in their slope are used in the correlation), a *contrast* across it, average and local *intensity* measures of the intervals to its left and right, 2-D *connectivity* to other edges on prior or subsequent lines, sub-pixel *positioning*, a measure of the extent of its *breadth*, and, increasing across the image, an *index*.


average intensity of interval to right
links to nearest neighbors
intensity locally to right
(sub-pixel) position
slope
intensity locally to left
average intensity of interval to left
edge index = i

Edge properties
Figure 1

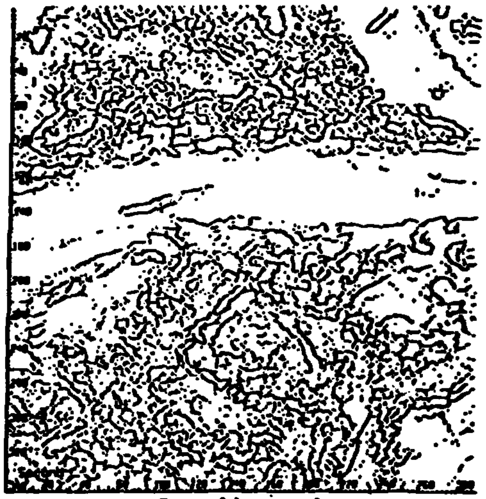
It is these properties of edges along scanlines in the two images which provide the metric for the correlation [1].



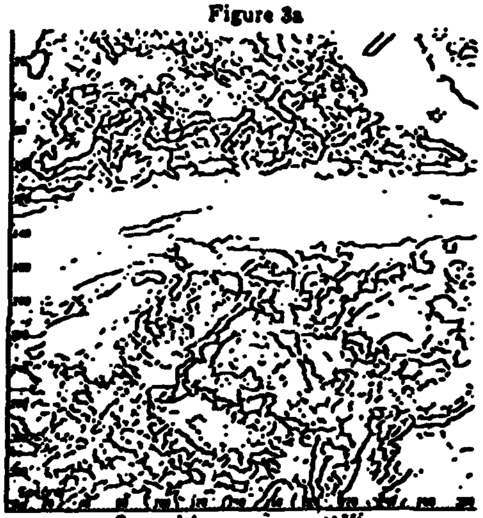
T's mark intensity values, vertical lines are edges
First and second image scanline edge depictions
Figure 2

Edge based descriptions of images are also generally more structured than are *area* based ones, as the linking phase of the process establishes edge connectivities in the two-dimensional image. This connection in the image plane, suggesting connectivity in the actual scene, provides the semantic component for a cooperative process to determine the best 3-D interpretation of the scene's edge depiction - the line-by-line edge correlation procedure chooses the best association of first image edges with second image edges from the available local information, then edge connectivity is used to either verify or repudiate these pairings.

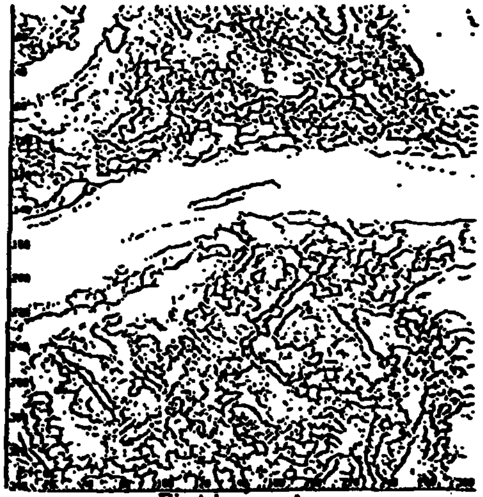
Experimentation was done with two basic approaches to the correlation, *contrast* based and *half-edge* based. This report will describe the genesis of the present system as it progressed through these approaches. Accompanying these changes in approach was a change in the basic computation process from a branch&bound search to a Viterbi [2] dynamic programming algorithm. This computation change was brought about by the realization that the analysis of busier images (such as the Night Vision Lab imagery), use of fewer ad hoc'isms in parameter settings (through measures of image statistics for finer control of noise based thresholds), and, as always, the search for greater generality, all lead to increased combinatorics. Although I will begin with a description of *contrast* based branch&bound, much of what is decribed applies to both correlation approaches and both methods of correlation computation.



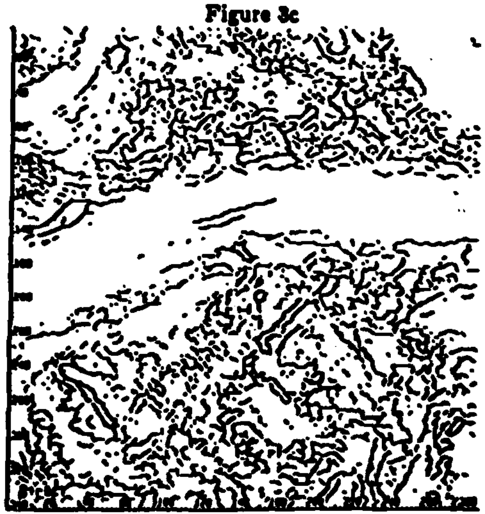
Second image edges
Figure 3a



Second image connectivity
Figure 3b



First image edges
Figure 3c



First image connectivity
Figure 3d

Preparing for the Correlation

Initially, for the branch&bound correlation, edges were grouped by their contrast sign and ordered by their strengths. An edge from the first image would be said to be a *possible mate* to an edge from the second if its *sign* was the same, their *strengths* similar, at least one of the surfaces bounding them to the left or the right being similar enough in *average intensity* in each view to be considered 'the same', and their relative positions *(indices)* being close enough that to associate them together would not exclude too many pairings from the correlation *(implied error)*. The insistence that the edge contrast signs be the same forbids the matching of an edge from say a grey surface projected against a white background with itself from a different view against a black ground (this restriction is not present in the *half-edge* based approach).

A list of such *possible mates* is then formed for each edge of the second image scanline and ordered by a sum of normalised scores of:

(difference in contrast)^2,

(difference in intensity of surfaces to sides)^2,

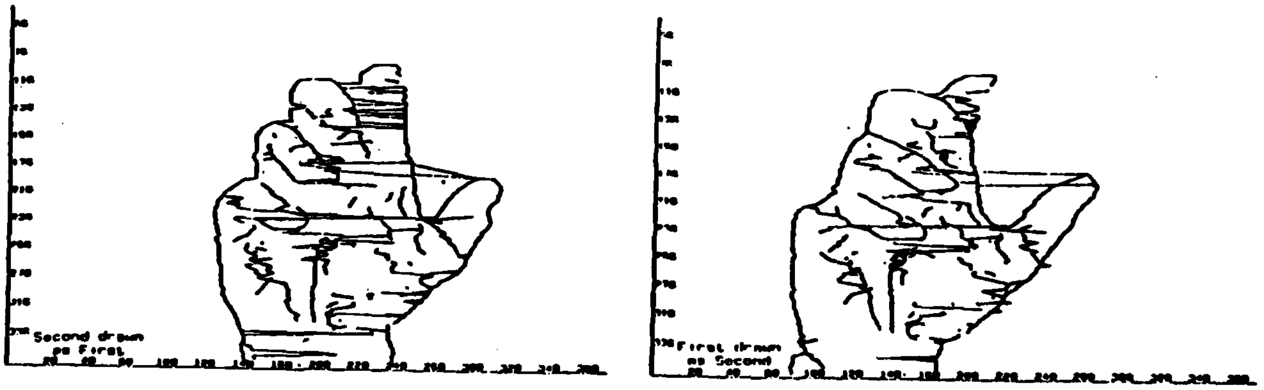
implied error (missed edges, if this association is made),

and a final nonlinear component putting edge matings whose left and right surfaces are *both* matchable to the head of the list. Correlation, then, is the process of finding the 'best' set of possible pairings of the edges.

Branch and Bound Correlation

What drives the correlation is a search for 'explanation' of the greatest number of edges in the scene (an 'explained' edge is one which has been unambiguously positioned in 3-space), and this metric provides the most effective element of the pruning technique (for indeed, the combinatorics can become far too great to allow unbounded expansion of the search tree). A running count of the *implied error* (edges which cannot be correlated) is maintained at each stage of the expansion of the correlation, and any partial assignment giving rise to an *implied error* above that of the 'best-so-far' (initially requiring 50% of edges to potentially match) is denied further expansion at that point.

This first approach, *contrast* based correlation, showed itself to be an acceptable technique with the data used for its development (Figure 4 demonstrates the quality of an early *contrast* based branch&bound correlation, when run with unrealistically high thresholds on a fairly noise-free non-busy stereo pair), but its shortcomings - not allowing contrast reversals to match on one side or another (such as grey moving from a white to a black ground), and requiring the actual step (contrast) to be of similar size - made it necessary to consider other approaches for a more general solution.



This depiction requires some explanation. The figure to the right is produced by following the connectivity in the first image, drawing lines between those edges in that image which have been associated by the correlation process with edges in the second image (ie. they have correlates in the other image), but rather than using the first image's coordinate references, the coordinate reference frame of the second image is used. This means that when following a connected set of edges in the first image, say the back of the hand, everything will look fine as viewed from the other image until an edge associated with something in the second image that's not the back of the hand is encountered. At this point a line will be drawn to whatever part of the second image that edge is associated with, producing a noticeable horizontal jag to that part of the image.

Correlation results
prior to cooperative connectivity enforcement
Figure 4

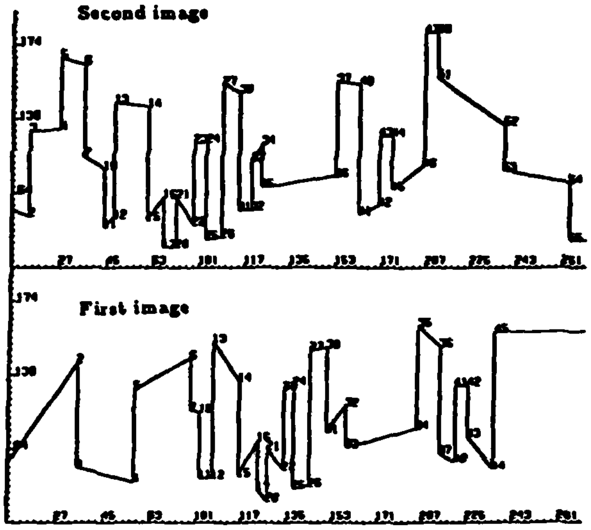
The second approach, *half-edge* based correlation, was developed to allow the possibility of contrast-reversed edges matching on one side or another. Here, the sign and strength of edges are ignored, and only the ordering parameters (as specified above) are used in *forming lists of possible mates* for each second image scanline edge. Unfortunately, there being in effect both a left and a right edge to deal with where before there was only one, the combinatorics tend to get way out of hand (a typical scanline in the Night Vision Lab imagery has about 30 edges, and the size of the search space on some of these was found to be up into the billions).

The combinatoric problem is not confined to *half-edge* based correlation, as in truth it is necessary with both approaches to limit the computation before letting the branch&bound correlation loose. This trimming was done, with both the *contrast* and the *half-edge based* approaches, by removing from consideration those scanline edges (either first or second image) of lowest contrast (where it is presumed that edges of lower contrast are less significant to the modelling) until the total estimable combinatorics passed under some prespecified maximum (50000 permutations). (*Implied error* bounding rarely let the actual number of permutations required exceed the hundreds).

Now, with either of the two approaches, the scanline correspondence having greatest number of 'explained' edges (in case of equivalent counts, lowest sum of squared intensity difference at sides of edges) is chosen as the result of the correlation. This is a set of pairs (see Figure 5):

$$\{ (f_i, s_j) \mid f_i \textit{ correlates with } s_j;$$
$$i, j \textit{ even} \rightarrow \textit{left of edge};$$
$$i, j \textit{ odd} \rightarrow \textit{right of edge} \}$$

Clearly there will be miscorrelations among these - experience, and a little thought, have shown that they can surely be expected to occur near the periphery of the scanline, where the need to correlate bounding edges is not present (the global constraint of maximising the edge pairings has diminishing effect near the sides, where the relative displacement of the images means no correlation exists). And of course there will always be edges that do look alike. With this local ambiguity, one can expect to always have incorrectly assigned edge pairs. It is up to a further analysis with more global information to remove these miscorrelations.

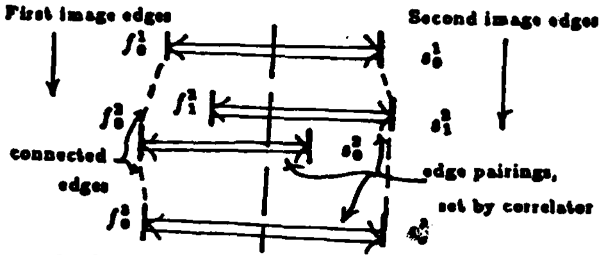


{ (2, 4) (5, 5) (6, 6) (7, 7) (10, 10) (11, 11) (12, 12) (13, 13) (14, 14) (15, 15) (16, 16) (17, 17) (21, 21) (22, 22) (23, 23) (24, 24) (25, 25) (26, 26) (27, 27) (30, 30) (35, 33) (36, 34) (37, 35) (41, 37) (42, 40) (43, 41) (44, 42) (45, 43) (46, 44) }

Set of associated pairs (as Figure 2)
f-first image, *s*-second image
Figure 5

Cooperative Continuity Enforcement

To the rescue comes a depth continuity enforcement process operating in a cooperative mode upon the edge pairings assigned by the correlation. It follows connected edges in the two image planes, removing those edge pairings that it finds to be inconsistent. An inconsistent pairing, in this sense, is one whose edges are nearest connected in 2-space (as seen in either image) to edges which have been paired differently by the correlation. This conflict in correlation is a necessary condition for inconsistency, but is not alone sufficient. For each pairing (f_i^m, s_j^m) on scanline m, and associated disparity δ_{ij}^m, measures η and σ are kept of the mean and standard deviation of changes in δ_{ij} among 2-space connected pairings. The other half of the consistency criterion is that the change in disparity $|\delta_{ij}^m - \delta_{pq}^n|$ from the pairing $(f_i^m, s_j^m)(\textit{disparity} = \delta_{ij}^m)$ to its nearest connected 2-space neighbour pair $(f_p^n, s_q^n)(\textit{disparity} = \delta_{pq}^n)$ be within $[\eta - \sigma, \eta + \sigma]$. A single such conflict is not enough to remove a pairing (as, really, which pairing is in error?). Rather, a pairing is only removed when it is found to be inconsistent from 2 different sources.



f_0^1, f_0^2, f_0^3 connected → (f_1^2, s_1^2) is bad

s_0^1, s_1^2, f_0^3 connected → (f_0^2, s_0^2) is bad

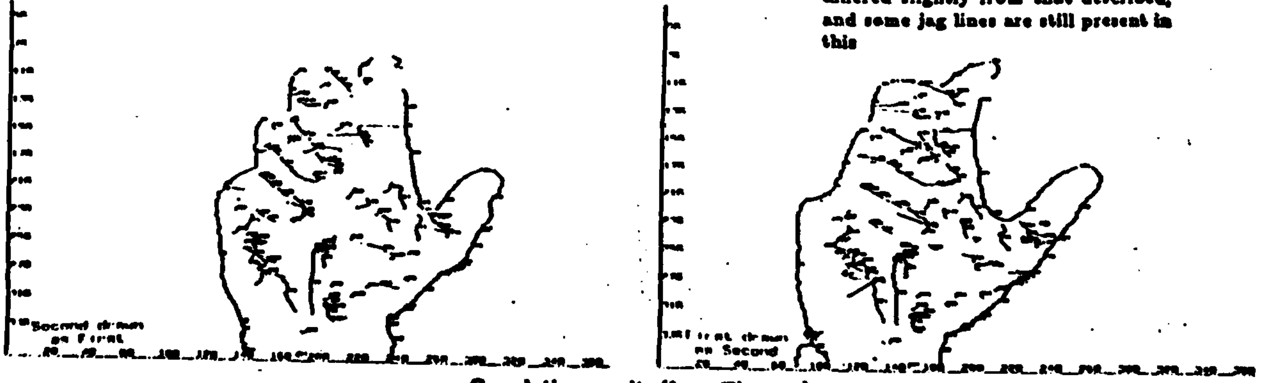
connectivity without consistent correspondence
Figure 6

When a correlation pairing (f_i^m, s_j^m) is removed, no immediate attempt is made to reassign the f_i^m or s_j^m (although it will be in the near future), and these edges are bypassed in the 2-space connectivity structure .. the paired edges above and below them are joined now as 2-space nearest connected pairs. The new change in disparity is evaluated, and tested to see whether it lies in the $[\eta - \sigma, \eta + \sigma]$ interval. When no further edge pairings can be removed by this process, all pairings left having a single inconsistency are removed. This more ruthless removal could not be applied earlier, as it would delete good pairings adjacent to bad ones in the process of removing the bad .

Certainly, a great deal more may be done with these edges unpaired by the removal process ... 2-D connectivity may make it clear where they should be really paired, the reduced combinatorics (generally, as fewer edges are left unpaired than began that way) may facilitate further correlation with relaxed constraints (particularly, allowing edge reversals - a left-right ordering in one image matching a right-left ordering in the other), etc. ... effort will be put in this direction later, once the correlation process itself is felt to be sufficiently stable and successful.

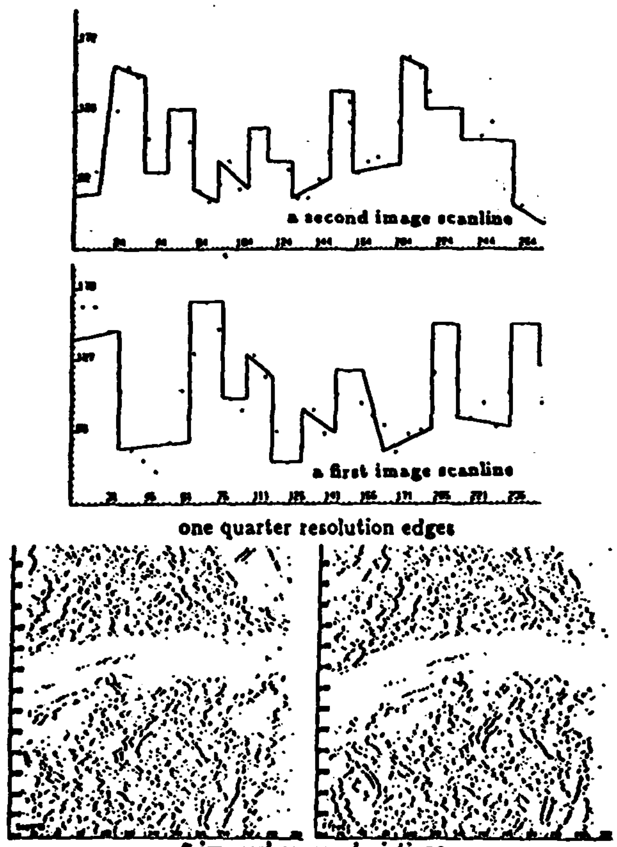
Coarse to Fine Correlation

A concern over the loss of potentially useful edge correlations through the combinatorial reduction then lead to a further change in the approach. In it *contrast* based and *half-edge* based correlation approaches are combined. It uses a resolution reduced 'planning' scheme that exploits the coarse image structure in reducing the correlation combinatorics. Here, the images are repeatedly halved in resolution with a 1-2-2-1 averaging operator (in effect removing the high frequency components, leaving the low-frequency, coarser structure more visible). The edges found at this resolution in the first and second images are treated as significant, or *landmark* edges, and are correlated in the *contrast* based edge style. For each set of assignments of these edges having *implied error* not greater than that of the 'best-so-far', the edges in the intervals between the *landmarks* are *half-edge* correlated. The reasons for this dual mode of operation are that: 1) it seemed a nice way to merge the two (equally valid and equally incomplete) edge matching assumptions, and 2) it was felt that the smoothed image edges would be more contrast preserving over viewpoint, having their high frequencies removed (this is a valid assumption for narrow to medium angle stereo, perhaps not for wide angle).



The cooperative algorithm used here differed slightly from that described, and some jag lines are still present in this

Correlation results (from Figure 4)
after cooperative connectivity enforcement
Figure 7



one quarter resolution edges

2 images' coarse depictions

Figure 8

There is always the danger with hierarchically constrained searches such as this that the best solution at a coarse level (reduced resolution) will not be consistent with the best overall solution at the finest level (full resolution). This problem was avoided in the branch&bound correlation by doing the full resolution interval correlation whenever the reduced resolution correlation suggested an improvement was possible over the best yet achieved. This recursive approach cannot be integrated into the Viterbi correlation as it stands, and this is one aspect of the Viterbi algorithm that we are looking into. Viterbi does, however, have some pretty outstanding advantages: it is optimal (when using the same assumption of no edge reversals in the images), polynomial (as opposed to exponential), and is very very fast.

Some recent results with the Viterbi algorithm follow. The principal complaints with these at the moment are the large number of apparently poor correlations (jag lines)they have, and the lack of good vertical connectivity in the image edge depictions. Our efforts to solve these problems are leading us to consider adding other local constraints for the edge matching, keeping alternative selections for each edge pairing, and examining forms of interpolatory correlation [4] in the surviving intervals defined by the correlation and cooperative consistency enforcement process. We have confidence that these will significantly improve the results. We would also like to work with more controlled images (remember the requirement that the epipolar lines here be parallel to the camera baseline, only in the CDC synthetic images was this the case), and our recent reinstallation of a pair of GE CCD cameras will help in this.

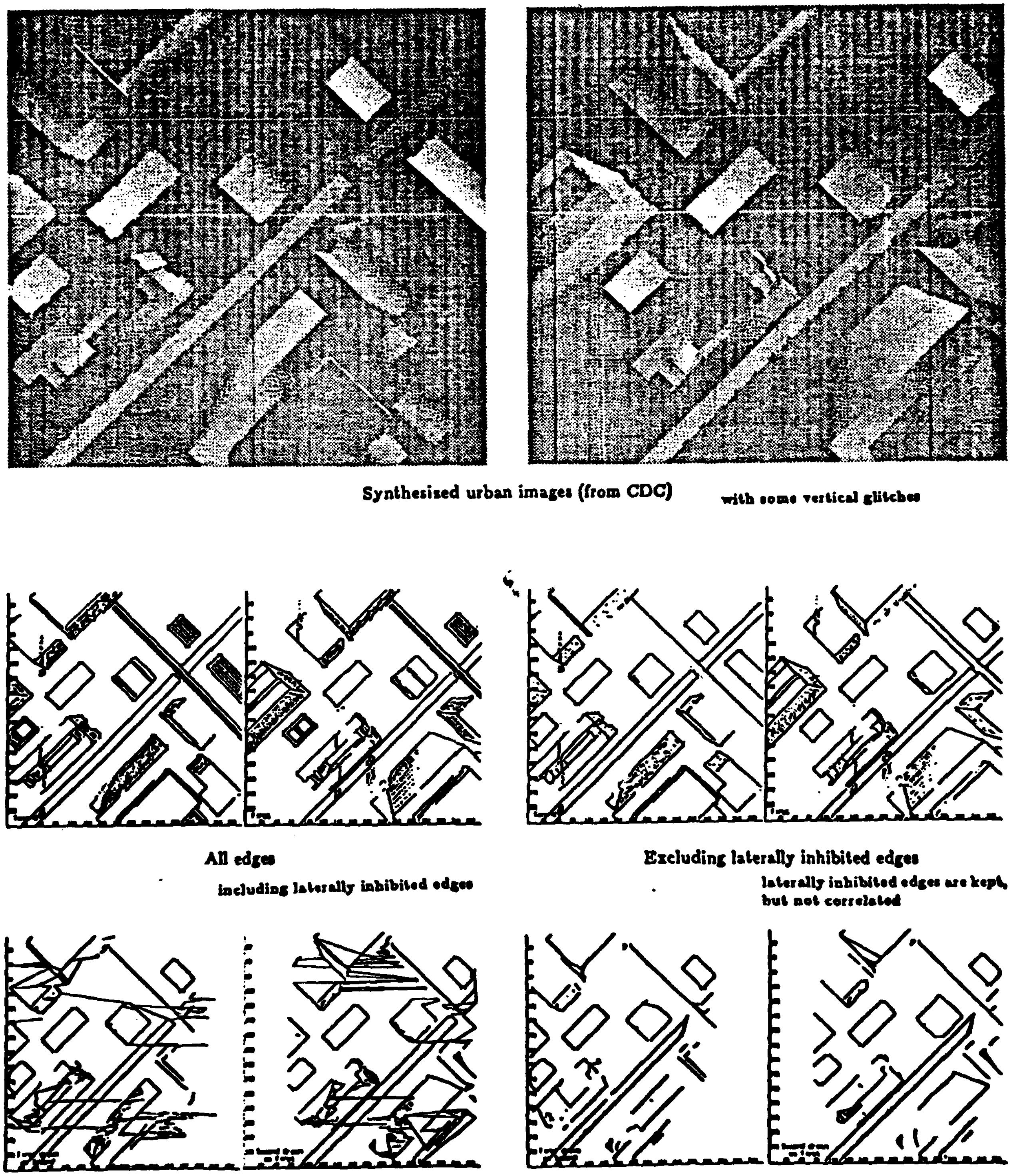
References

[1] Arnold, R. David , 'Local Context in Matching Edges for Stereo Vision,' *Proc. ARPA Image Understanding Workshop*, Cambridge, Mass. May 1978, 65-72.

[2] Forney, G. David Jr., 'The Viterbi Algorithm,' *Proc. IEEE*, Vol. 61, No. 3, March 1973.

[3] Henderson, Robert L., Walter J. Miller, C. B. Grosch, 'Automatic Stereo Recognition of Man-Made Targets,' *Soc. Photo-Optical Instrumentation Engineers*, Vol. 186, August 1979.

[4] Panton, Dale J., 'A Flexible Approach to Digital Stereo Mapping,' *Photogrammetric Engineering and Remote Sensing*, Vol. 44, No. 12, December 1978, 1499-1512.



Synthesised urban images (from CDC) with some vertical glitches

All edges

including laterally inhibited edges

Excluding laterally inhibited edges

laterally inhibited edges are kept, but not correlated

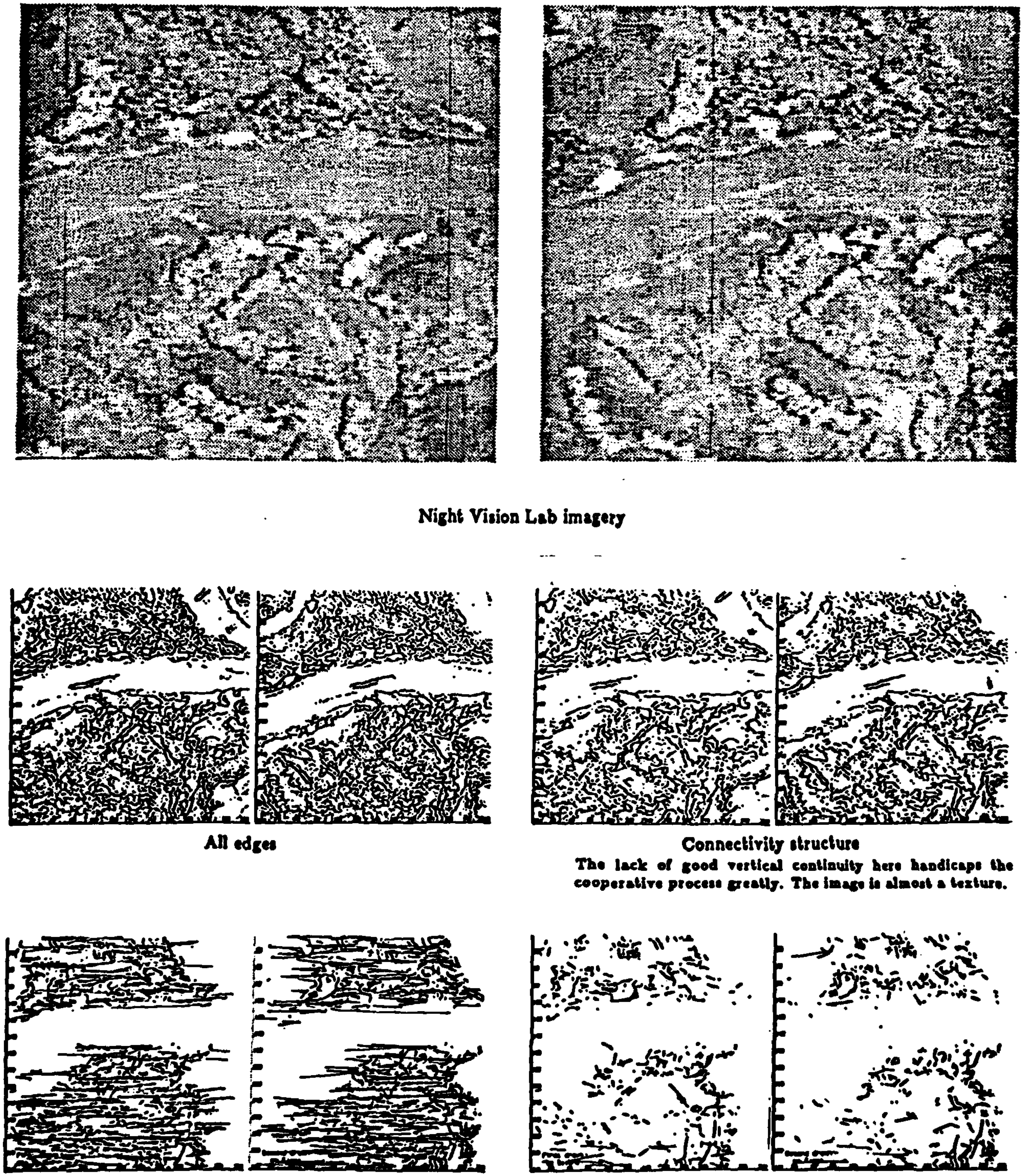
After correlation

Left side of edge correlations are superimposed with right side of edge correlations

Figure 9

After cooperative process

3900 half-edge pairings (57% of possible edges) 7 sec. first image, 12 sec. second image plus correlation. 37% removed by cooperative process. .



Night Vision Lab imagery

All edges

Connectivity structure

The lack of good vertical continuity here handicaps the cooperative process greatly. The image is almost a texture.

After Correlation

After cooperative process

Figure 10 5300 half-edge pairings (65%of possible edges) 5 sec. first image, 15 sec. second image plus correlation. 43%removed by cooperative process.